This document is based on European standards and is not valid for use in U.S.A.

Distributed / AS-Interface / Logic Controller / M238

System User Guide

EIO0000000281

MAR 2010

Schneider Electric

Contents

Important Information 3
Before You Begin 4
Introduction 6
Abbreviations 7
Glossary 8
Application Source Code 9
Typical Applications 10
System 11
Architecture 11
Installation 14
Hardware 19
Software 27
Communication 28
Implementation 33
Communication 35
Controller 39
HMI 72
AS-Interface device addressing 86
Safety Monitor 92
Appendix 107
Detailed Component List 107
Component Protection Classes 110
Component Features 111
Contact 115

Important Information

NOTICE Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation, which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation, which, if not avoided, **can result in** death, serious injury, or equipment damage.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation, which, if not avoided, **can result in** injury or equipment damage.

PLEASE NOTE Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and the installation, and has received safety training to recognize and avoid the hazards involved

© 2008 Schneider Electric. All Rights Reserved.

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to **the operator of that machine.**

⚠ WARNING
UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY • Do not use this software and related automation products on equipment which does not have point-of-operation protection. • Do not reach into machine during operation. **Failure to follow these instructions can cause death, serious injury or equipment damage.**

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation and maintenance of the machine; therefore, only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to the applicable local and national standards and regulations. A "National Safety Council's" Accident Prevention Manual also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products by itself cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks for point-of-operation protection have been installed and are operational before placing the equipment into service. All mechanical/electrical interlocks and safeties for point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of this document.

START UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters and debris from equipment.

Failure to follow these instructions can result in injury or equipment damage.

Follow all start up tests recommended in the equipment documentation. Store all equipment documentation for future reference.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove ground from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and rating of components, there are hazards that can be encountered if such equipment is improperly operated.

- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.

- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

⚠ WARNING

UNEXPECTED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can cause death, serious injury or equipment damage.

Introduction

Introduction

This document is intended to provide a quick introduction to the described system. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers additional information to the product documentation, for installing, configuring and implementing the described system.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider-Electric and third-party components may be integrated to fulfill an industrial application.

A detailed functional description or the specification for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The architecture described in this document has been fully tested in our laboratories using all the components found in the Component List at the end of this document. Of course, your specific application requirements may be different and naturally may require additional or different components. In this case, you will have to adapt the information provided in this document to your particular needs. To do so, you will need to consult the specific product documentation of the components that you are substituting in this architecture. Pay particular attention in conforming to any safety information, different electrical requirements and normative standards that apply to your adaptation.

It should be noted that there are some major components in the architecture described in this document that cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified herein. You must be aware of the consequences of component substitution in the architecture described in this document as substitutions may impair the compatibility and interoperability of software and hardware.

⚠ CAUTION

EQUIPMENT INCOMPATIBILITY OR INOPERABLE EQUIPMENT

Read and thoroughly understand all hardware and software documentation before attempting any component substitutions.

Failure to follow these instructions can result in injury or equipment damage.

Abbreviations

Abbreviation	Signification
AC	Alternating Current
CB	Circuit Breaker
CFC	Continuous Function Chart – a programming language based on function chart
DI	Digital Input
DO	Digital Output
DC	Direct Current
DFB	Derived Function Blocks
EDS	Electronic Data Sheet
E-STOP	Emergency Stop
FBD	Function Block Diagram – an IEC-61131 programming language
HMI	Human Machine Interface
I/O	Input/Output
IL	Instruction List - a textual IEC-61131 programming language
LD	Ladder Diagram – a graphic IEC-61131 programming language
PC	Personal Computer
POU	Programmable Object Unit, Program Section in SoMachine
PS	Power Supply
RMS	Root Mean Square
RPM	Revolutions Per Minute
SE	Schneider Electric
SFC	Sequential Function Chart – an IEC-61131 programming language
ST	Structured Text – an IEC-61131 programming language
TVDA	Tested, Validated and Documented Architecture
VSD	Variable Speed Drive
WxHxD	Dimensions : Width, Height and Depth

Glossary

Expression	Signification
Advantys	SE product name for a family of I/O modules
AS-Interface	Name for a communications machine bus system
ASWIN	SE product name for the Safety Monitor configuration software
Harmony	SE product name for a family of switches and indicators
Magelis	SE product name for a family of HMI-Devices
Phaseo	SE product name for a family of power supplies
Preventa	SE product name for a family of safety devices
Safety Monitor	SE product name for a device supervising the functional safety of the system
SoMachine	SE product name for an integrated software tool
TeSys	SE product name for motor starters and load contactors
Vijeo Designer	An SE software product for programming Magelis HMI devices

Application Source Code

Introduction

Examples of the source code and wiring diagrams used to attain the system function as described in this document can be downloaded from our website.

The example source code is in the form of configuration, application and import files. Use the appropriate software tool to either open or import the files.

Extension	File Type	Software Tool Required
AS2	Configuration file	ASWIN software
DOC	Document file	Microsoft Word
PDF	Portable Document Format - document	Adobe Acrobat
PROJECT	Project file	SoMachine
Z13	Project file	EPLAN

Typical Applications

Introduction

Here you will find a list of the typical applications, and their market segments, where this system or subsystem can be applied:

Building / services

- Access and entry control automated systems (Door, awning, roller blind..)

Lift

- Escalator

System

Introduction

The system chapter describes the architecture, the dimensions, the quantities and different types of components used within this system.

Architecture

General

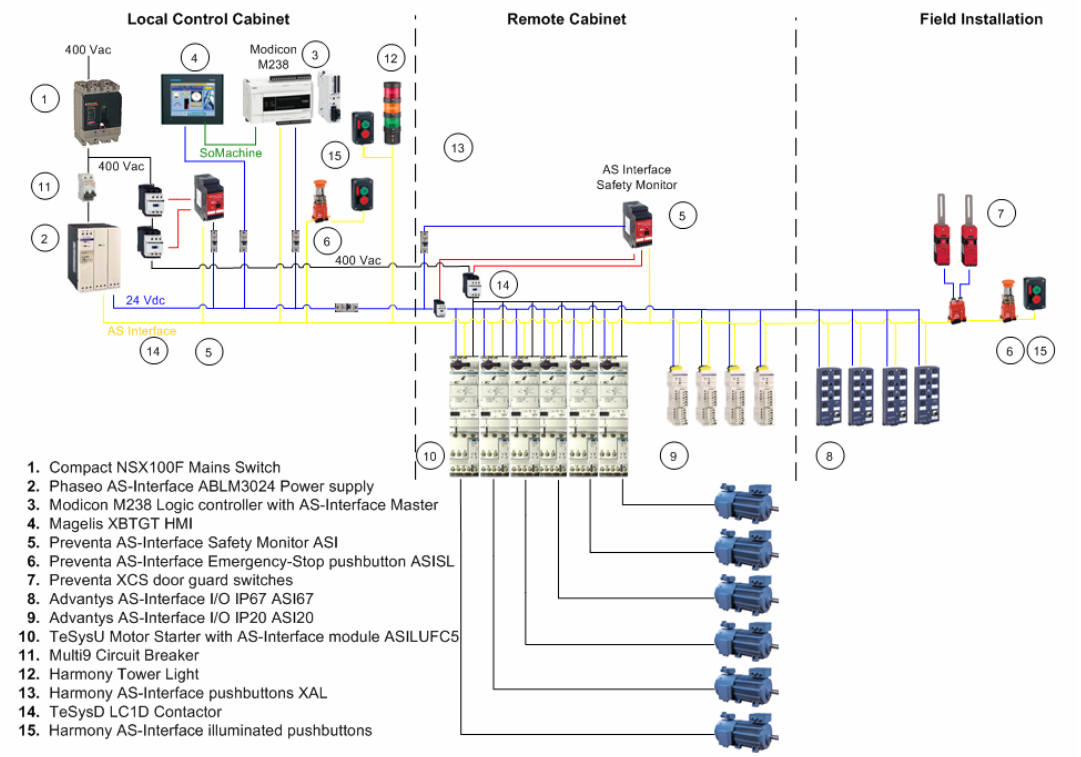
The controller in this application is a M238 Logic controller. The user controls the application using the Magelis HMI.

The example application includes two functional safety options according to IEC 61508 standards: an Emergency Stop function supervised by a Preventa ASISAFEMON1 Safety Monitor (see the appropriate hardware manual), plus a second Preventa Safety Monitor to evaluate protective door sensors.

The system consists of a control cabinet with the operator interface, a remote cabinet with the motor control and a subsystem for the field installation.

The System includes an AS-Interface bus for communication between different devices.

Layout


Local Control Cabinet
Remote Cabinet
Field Installation
400 Vac
Modicon M238
SoMachine
400 Vac
AS Interface Safety Monitor
400 Vac
24 Vdc
AS Interface
1. Compact NSX100F Mains Switch
2. Phaseo AS-Interface ABLM3024 Power supply
3. Modicon M238 Logic controller with AS-Interface Master
4. Magelis XBTGT HMI
5. Preventa AS-Interface Safety Monitor ASI
6. Preventa AS-Interface Emergency-Stop pushbutton ASISL
7. Preventa XCS door guard switches
8. Advantys AS-Interface I/O IP67 ASI67
9. Advantys AS-Interface I/O IP20 ASI20
10. TeSysU Motor Starter with AS-Interface module ASILUFC5
11. Multi9 Circuit Breaker
12. Harmony Tower Light
13. Harmony AS-Interface pushbuttons XAL
14. TeSysD LC1D Contactor
15. Harmony AS-Interface illuminated pushbuttons

Components

Hardware:

- Main switch type Compact NSX100
- Modicon M238 Logic controller with AS-Interface Master TWDNOI10M3
- Magelis XBTGT 2220 HMI
- TeSysD load contactor LC1D
- Power supply Phaseo AS-Interface ABL M3024
- Preventa AS-Interface Safety Monitor ASISAFEMON1
- Preventa door guard switches with rotary release (trigger action)
- Multi 9 circuit breakers
- TeSysU with AS-Interface module ASILUFC5
- Harmony AS-Interface indicator bank
- AS-Interface Pushbuttons XALD
- AS-Interface Pushbuttons XB5
- IP20 I/O digital modules with AS-Interface (ASI20MT)
- IP67 I/O digital modules with AS-Interface (ASI67FMP)

Software:

- SoMachine V2.0
- ASIWIN2 V2.03

Control Cabinet

The control cabinet comprises the following main components:

- Modicon M238 Logic controller with AS-Interface Master TWDNOI10M3
- Magelis XBTGT 2220 HMI
- Preventa AS-Interface Safety Monitor, AS-Interface emergency-stop switch with redundant load contactors for disconnecting the 400 V AC power supply

The central devices in this control cabinet include the M238 logic controller, the Magelis HMI and the optional functional safety components (per IEC 61508). The functional safety components work together in the event of an emergency-stop condition. The load is disconnected via a Preventa AS-Interface Safety Monitor with assigned contactors.
In addition, the control or main cabinet contains the main switch for the motor components (400/230 Vac), as well as the conventional 24 Vdc and AS-Interface power supplies.

Remote Cabinet

The remote cabinet comprises the following main components:

- 6 TeSysU motor starters with AS-Interface module ASILUFC5
- 4 digital IP20 I/O modules with AS-Interface (ASI20MT)
- 4 digital IP67 I/O modules with AS-Interface (ASI67FMP)
- AS-Interface safety monitor, AS-Interface emergency-stop switch with redundant load contactors for disconnecting the 400 Vac power supply.

The TeSysU motor starters can be installed side-by-side and supplied with power via bus bars in order to minimize requirements in respect of space and wiring. The IP20 I/O modules can also be installed side-by-side to save space. An AS-Interface safety monitor, with associated redundant load contactors for implementing an optional safety guard function, is also found in this cabinet.

Optional Safety Function

The AS-Interface emergency-stop switch is connected to the shared yellow AS-Interface cable (this means that it *does not have to be wired separately*). Although the emergency-stop switches on the main cabinet and on the remote cabinet trigger both safety monitors, if the door safety function (which is controlled by the roller limit switches) is cancelled; only the safety monitor in the remote cabinet is disconnected.

Field Installation

The field installation essentially comprises of four ASI67FMP43E digital IP67 I/O modules. As the IP67 modules and their connection adapters do not require enclosures, their installation location and type can be selected in accordance with the requirements. Field installations can, therefore, be set up anywhere, and in any configuration, using cables and sensors with the appropriate degree of protection.

Quantities of Components

For a complete and detailed list of components, the quantities required and the order numbers, please refer to the components list at the rear of this document.

Degree of Protection

Not all of the components in this configuration are designed to withstand the same environmental conditions. Some components may need additional protection, such as housings, depending on the environment in which you intend to use them. For environmental details of the individual components please refer to the list in the appendix of this document and the corresponding user manual.

Cabinet Technical Data

Input	Mains voltage	400Vac
	Power requirement	~ 3 kW
	Cable Size	5 x 2.5 mm² (L1, L2, L3, N, PE)
	Cable connection	3 phase + Neutral + Ground Neutral is needed for 230 Vac (Phase and Neutral)
Output	Motor power ratings	6 asynchronous motors 0.18 kW (4 poles:1500 RPM)

Functional Safety Notice

(EN ISO 13849-1 EN IEC 62061)

The standard and level of functional safety you apply to your application is determined by your system design and the overall extent to which your system may be a hazard to people and machinery.

As there are no moving mechanical parts in this application example, category 3 (according to EN ISO 13849-1) has been selected as an optional safety level.

Whether or not this functional safety category should be applied to your system should be ascertained with a proper risk analysis.

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the functional safety requirements with respect to the equipment used in their systems or of compliance with either national or international safety laws and regulations

Emergency Stop

Emergency Stop/Emergency Disconnection function

This function for stopping in an emergency is a protective measure which compliments the safety functions for the safeguarding of hazardous zones according to prEN ISO 12100-2.

Safety Function

Door guarding

up to Performance Level (PL) = c, Category 3, Safety Integrity Level (SIL) = 2

Dimensions

The dimensions of the individual devices used; controller, power supply, etc. require a housing cabinet size of at least 800 x 600 x 300 mm for control cabinet and 600 x 600 x 300mm for the remote cabinet (WxHxD).

The HMI display, illuminated indicators such as “SYSTEM ON“, “SYSTEM OFF“ or “ACKNOWLEDGE EMERGENCY STOP“ as well as the emergency off switch itself, can be built into the door of the cabinet.

Installation

Introduction

This chapter describes the steps necessary to set up the hardware and configure the software required to fulfill the described function of the application.

Layout

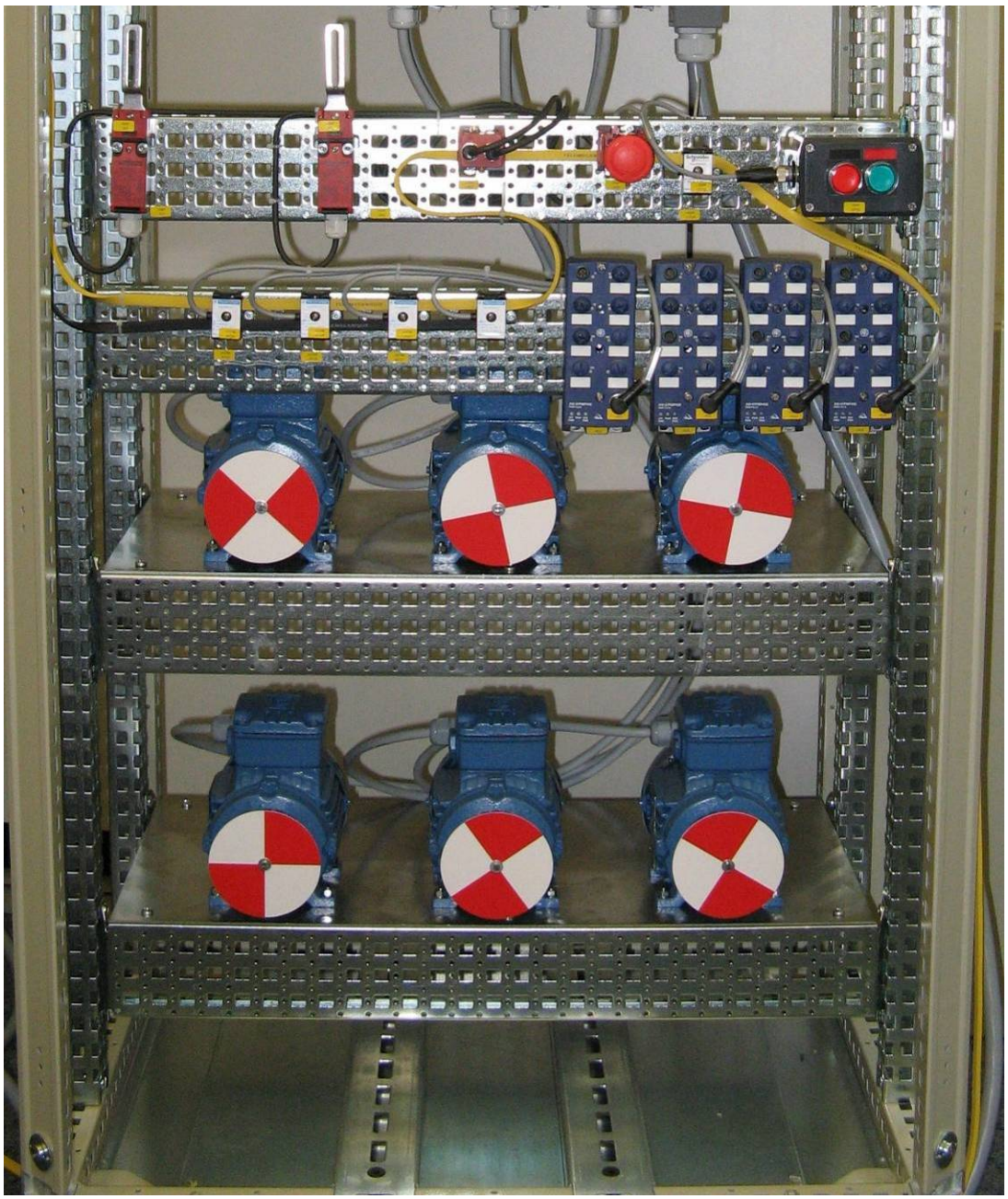
Main Cabinet front

Main Cabinet interior

Remote Cabinet front

Remote Cabinet interior

Field installation



Notes

The components designed for installation in a cabinet, i.e. the controller, safety modules, circuit breakers, contactors, motor circuit breakers, power supply, TeSysU motor starters and M238 extension modules can be mounted on a 35 mm DIN rail.

The master switch is screwed directly onto the mounting plate.

The complete field installation and the harmony indicator bank are designed for on-wall mounting in the field. The XB5 pushbuttons in XALD housing are designed for backplane assembly or direct wall mounting.

The emergency-off switches and the installation box for the main switch have been designed for installation in a cabinet door.

400 Vac 3-phase wiring between the main circuit breaker, motor circuit, motor starters and motors.

230 Vac single phase wiring between the main circuit breaker and primary side of the 24V power supply.

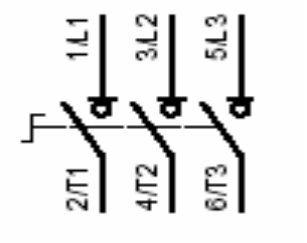
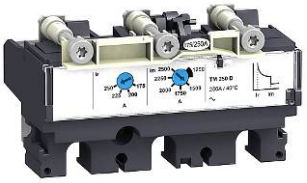
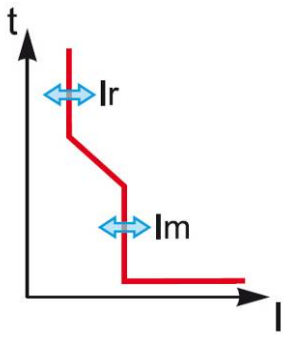
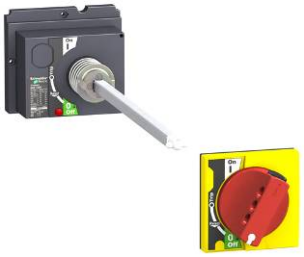
24 Vdc wiring for control circuits and the controller, I/O modules and the HMI power supply.

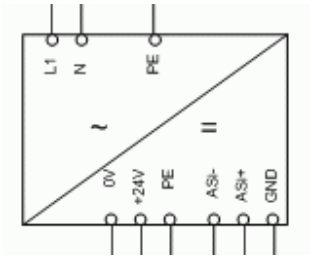
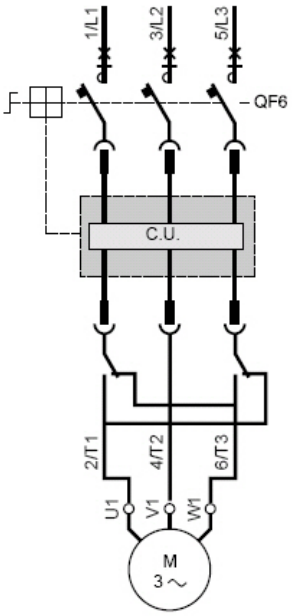
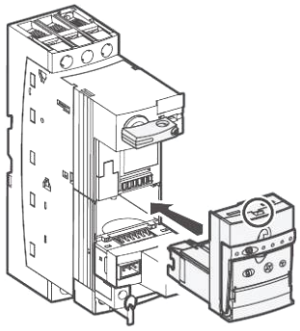
AS-Interface line (30 Vdc) and auxiliary power supply (24 Vdc) via yellow and black AS-Interface cable.

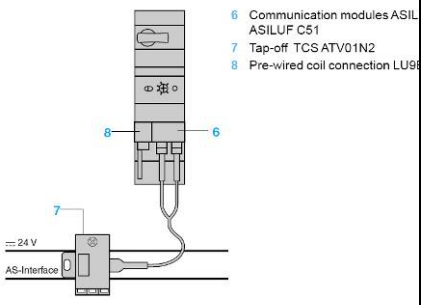
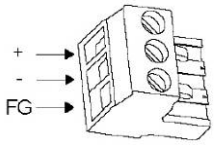
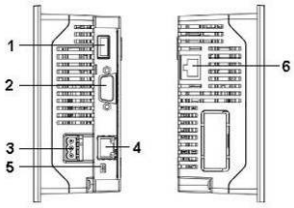
The individual components must be interconnected in accordance with the detailed circuit diagram in order to ensure that they function correctly.

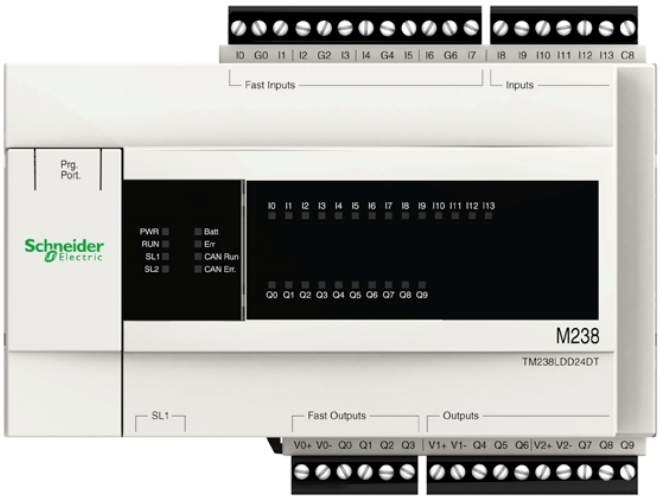
Serial line cable is installed for the communication link between the controller and the HMI.

Hardware

Main Switch **Compact NSX100F** **LV429003** **36 kA 380/415 Vac**		
Main Switch **Compact NSX100F** **LV429035** Trip unit TM32D Thermal-magnetic 32 A		 Ir - Thermal protection Im - Magnetic protection
Main Switch **Compact NSX100F** **Rotary handle LV429340** **Terminal shield LV429515**	 Rotary handle with red handle on yellow front	 Terminal shield short
Multi9 **circuit breaker 2-pole** **23746**		

Multi9 **Circuit Breaker 1-pole** **23726**		
Phaseo **Power Supply unit** **ASIABLM3024**		
TeSysU **Motor Starter** **Power section, (with reversing contactor)** **LU2B12B** **and** **Control module** **LUCA05BL**		
TeSysU **Motor Starter** **Control Unit** **LUCA05BL**		

TeSysU **Motor Starter** **Coil Unit** **TeSysU** **LU9MRL**		
TeSysU **Motor Starter** **AS-Interface module** **ASILUFC5**		
Magelis **Graphic HMI** **XBTGT2220**	  + 24Vdc - 0 V FG Ground	 1 USB port (USB 1.1) 2 COM1 serial port (SubD, 9-pin) 3 Current input terminal block (see image on left) 4 COM2 serial port (RJ45) 5 Polarity selector switch 6 Ethernet interface (optional, XBTGT2330)

Modicon M238 **Logic controller** **TM238LDD24DT** 14 Digital Input 10 Digital Output	

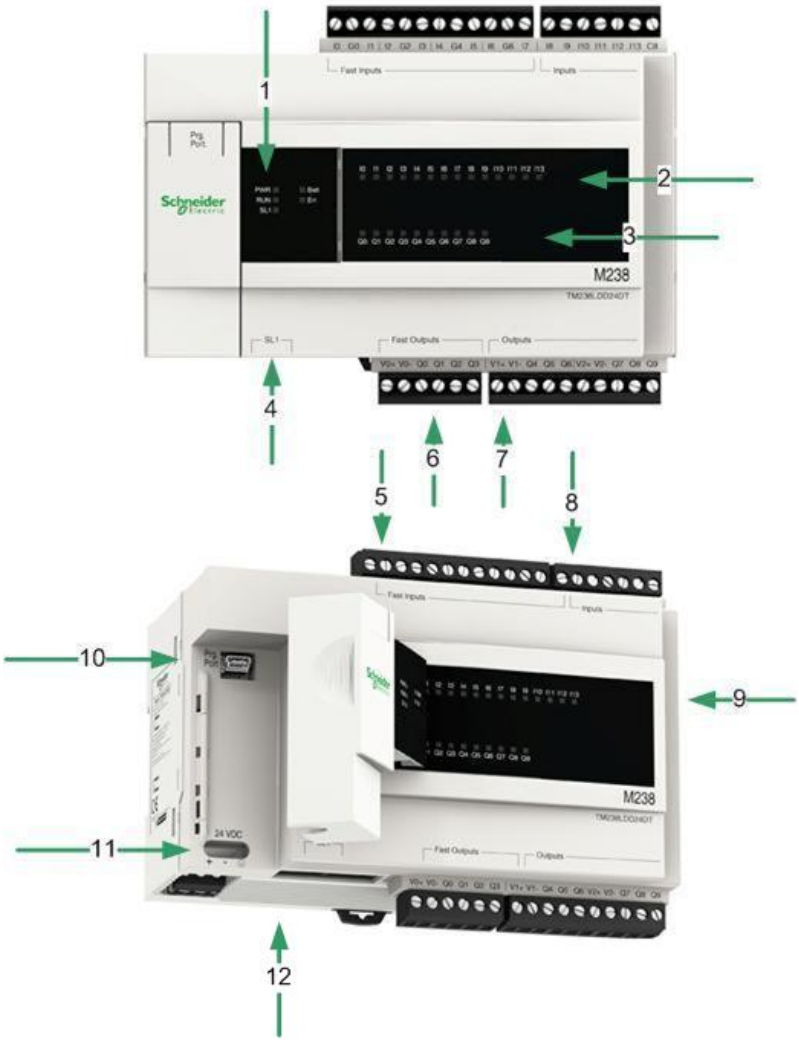
Modicon M238

Logic controller

TM238LDD24DT

14 Digital Input

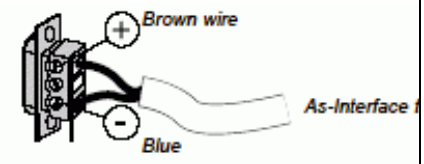
10 Digital Output



1. The controller and integrated communication port status by means of 5 LEDs (PWR, RUN, Batt, Err and SL1).
2. A display unit showing the Input states (I0…I13).
3. A display unit showing the Output states (Q0…Q9).
4. RJ 45 connector for connection of a serial link marked SL1 (SoMachine protocol).
5. A removable screw terminal block (12 terminals) for connecting the sensors (24 Vdc fast inputs).
6. A removable screw terminal block (6 terminals) for connecting the 4 pre-actuators (24 Vdc fast outputs).
7. A removable screw terminal block (10 terminals) for connecting the 6 pre-actuators (24 Vdc outputs).
8. A removable screw terminal block (7 terminals) for connecting the sensors (24 Vdc inputs).
9. A connector for extension modules, for example TWDNOI10M3 (7 modules max.).
10. Mini B USB Port, for a programming terminal.
11. A non-removable screw terminal block (3 terminals +, -, t marked 24 Vdc) to connect the 24 Vdc power supply. With access from the bottom of the controller:
12. A hinged cover for accessing the optional backup battery for the RAM memory andthe real-time clock inside the base.

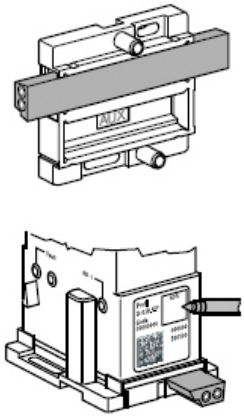
<table>
<tr>
<td>Twido

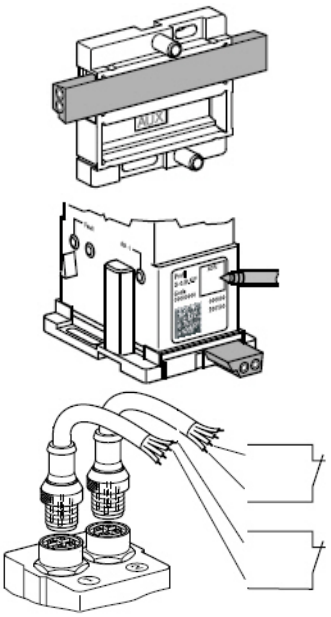
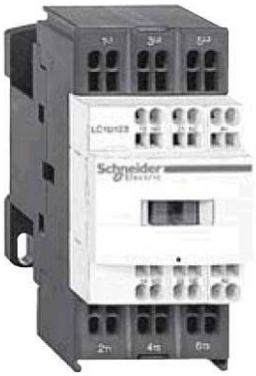
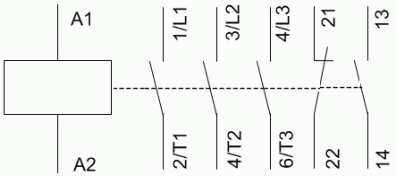
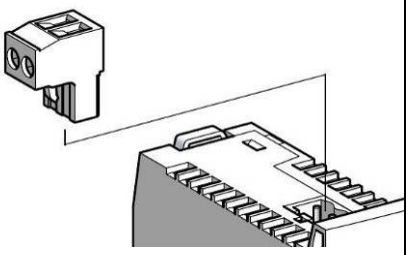
AS-Interface master module for M238

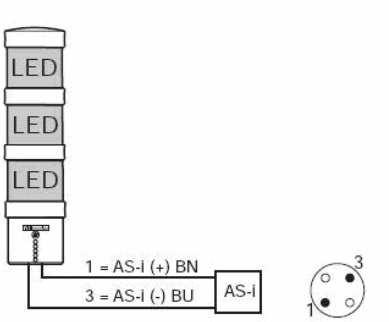
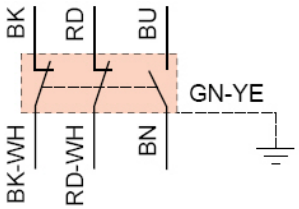
TWDNOI10M3</td>
<td> </td>
<td> 
</td>
</tr>
<tr>
<td>Preventa

AS-Interface safety monitor

ASISAFEMON1</td>
<td> </td>
<td>
<table>
<tr><th colspan="2">Terminal</th><th rowspan="2">Signal / description</th></tr>
<tr><th>ASISAFE MON1(B)</th><th>ASISAFE MON2(B)</th></tr>
<tr><td>AS-i+</td><td>AS-i+</td><td rowspan="2">Connection at the AS-interface bus</td></tr>
<tr><td>AS-i–</td><td>AS-i–</td></tr>
<tr><td>L+</td><td>L+</td><td>+24 V ⎓ / supply voltage</td></tr>
<tr><td>M</td><td>M</td><td>GND / reference ground</td></tr>
<tr><td>FE</td><td>FE</td><td>Functional earth</td></tr>
<tr><td>1.Y1</td><td>1.Y1</td><td>EDM 1 / input of external device monitoring circuit, output circuit 1</td></tr>
<tr><td>1.Y2</td><td>1.Y2</td><td>Start 1 / start input, output circuit 1</td></tr>
<tr><td>1.13</td><td>1.13</td><td rowspan="2">Output switching element 1, output circuit 1</td></tr>
<tr><td>1.14</td><td>1.14</td></tr>
<tr><td>1.23</td><td>1.23</td><td rowspan="2">Output switching element 2, output circuit 1</td></tr>
<tr><td>1.24</td><td>1.24</td></tr>
<tr><td>1.32</td><td>1.32</td><td>Message output 1 “Safety on”, output circuit 1</td></tr>
<tr><td>–</td><td>2.Y1</td><td>EDM 2 / input of external device monitoring circuit, output circuit 2</td></tr>
<tr><td>–</td><td>2.Y2</td><td>Start 2 / start input, output circuit 2</td></tr>
<tr><td>–</td><td>2.13</td><td rowspan="2">Output switching element 1, output circuit 2</td></tr>
<tr><td>–</td><td>2.14</td></tr>
<tr><td>–</td><td>2.23</td><td rowspan="2">Output switching element 2, output circuit 2</td></tr>
<tr><td>–</td><td>2.24</td></tr>
<tr><td>–</td><td>2.32</td><td>Message output 2 “Safety on”, output circuit 2</td></tr>
</table>
</td>
</tr>
<tr>
<td>Emergency-Stop

ASI SSL B4 + ZB4BS844</td>
<td> </td>
<td>  </td>
</tr>
</table>

Safety Input Slave **1 x 2 inputs** **ASISSLC2**		
TeSysD **Load Contactor** **LC1D09BL**		
AS-Interface Input/output module (IP20) **ASI20MT4I3OSE with ASI20MACC4**		 **ASI 20MACC4**

AS-Interface input/output block IP67 **ASI67FMP43E with TCSATV011F1**		 **TCSATV011F1**
Harmony **Tower Light** **XVBC**		
Safety limit switch **XCSM3915L1**		

Software

General

The main programming work lies in the programming of the M238, the configuration of the AS-Interface and creating the screens for the HMI display.

The Modicon M238 Logic controller is programmed using SoMachine.

The HMI application on the Magelis XBTGT 2220 display is created using Vijeo Designer.

The safety monitor application is programmed with ASIWIN2.

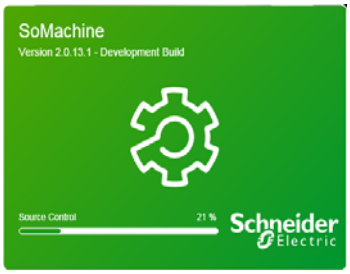
To use the software packages, your PC must have the appropriate Microsoft Windows operating system installed:

- Windows XP Professional

The software tools have the following default install paths:

SoMachine:

C:\Program files\Schneider Electric\SoMachine



Vijeo-Designer (included in SoMachine):

C:\Program files\Schneider Electric\VijeoDesigner

ASIWIN2:

C:\Program files\Schneider Electric\ASIWIN

Communication

General

This architecture uses 2 different communication networks:

- SoMachine protocol
- AS-Interface

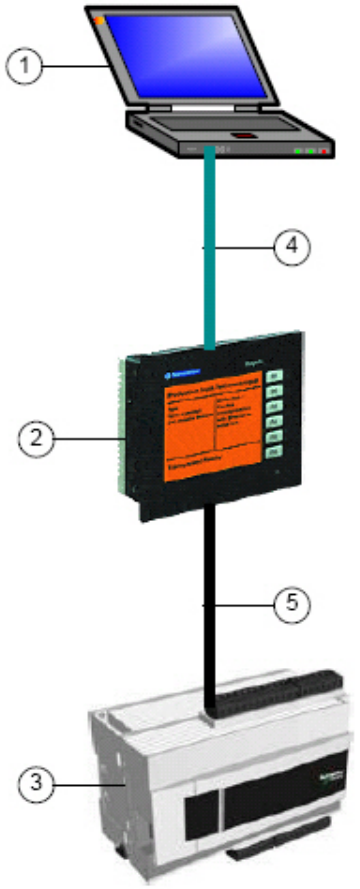
The described architecture includes two different communication busses. The AS-Interface includes the Modicon M238 Logic controller with AS-Interface Master, Advantys ASI-I/O + FTB I/O-Islands and TeSysU motor starters. All the devices and the I/O-Islands are connected to the AS-Interface via AS-Interface TAPs.

The Modicon M238 Logic controller and the Magelis HMI communicate via SoMachine protocol on RS485. The download from the PC to the M238 and to the HMI is done with a single connection. The PC has to be connected to the HMI and this connection is also used to send data to the M238.

PC ↔ XBTGT ↔ M238

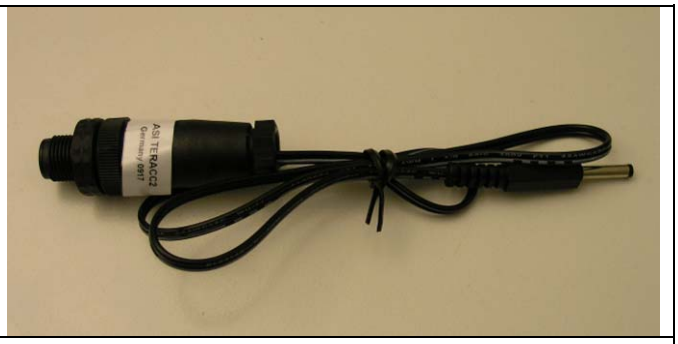
The download direction is from the PC to the HMI and via the HMI to the M238.

Note:
For a direct connection of the PC to the controller the cable TCSXCNAMUM3P can be used.

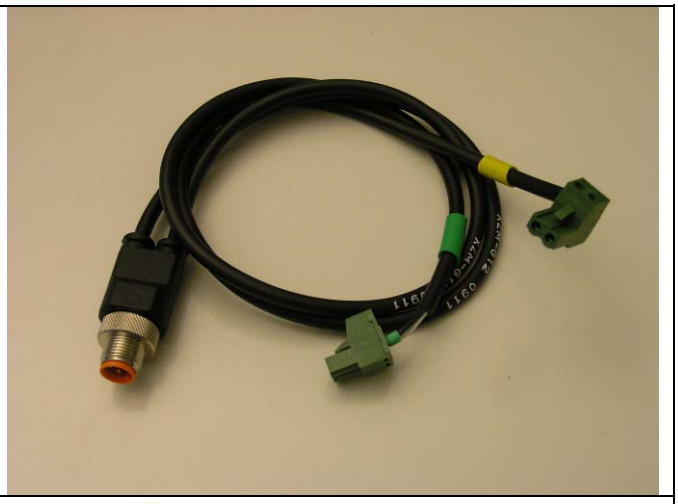


1. PC
2. Magelis XBTGT
3. Modicon M238
4. USB to USB cable **XBTZG935** (SoMachine protocol)
5. SubD9 to RJ45 cable **XBTZ9008** (SoMachine protocol)

SubD9 to RJ45 cable **XBTZ9008** **HMI ↔ Controller**	
USB to USB cable **XBTZG935** **PC ↔ HMI**	
AS-Interface **Master module** **TWDNOI10M3** **Expansion module** **connects to the M238**	
AS-Interface **IP20 In-/Output modules for use in a cabinet** **ASI 20MT4I3OSE**	

ASI 20MT4I3OSE **addressing cable** **ASITERACC2** **M12→chinch**	
AS-Interface IP67 Input / Output modules for use in the field **ASI67FMP43E**	
ASI67FMP43E **addressing cable** **ASITERACC1F** **M12F→ M12M**	
AS-Interface Safety Monitor **ASISAFEMON1**	

ASISAFEMON1 **programming cable ASISCPC** **SubD9 to RJ45 cable** **PC ↔ ASISAFEMON1**	
AS-Interface Safety slaves **for implementing safety functions :** **ASISSLB2** **ASISSLC2**	 ASISSLB2 ASISSLC2
ASISSLB2 **ASISSLC2** **addressing cable** **ASITERR1** **M12 → IR**	
AS-Interface **module ASILUFC5**	

ASILUFC5 **addressing cable** **XZM-G12** **M12→ 2pol green**	
AS-Interface cable	 Yellow AS-Interface Cable 2 x 1.5 mm^2 Black Auxiliary Cable 2 x 1.5 mm^2

Implementation

Introduction

The implementation chapter describes all the steps necessary to initialize, to configure, to program and start-up the system to achieve the application functions as listed below.

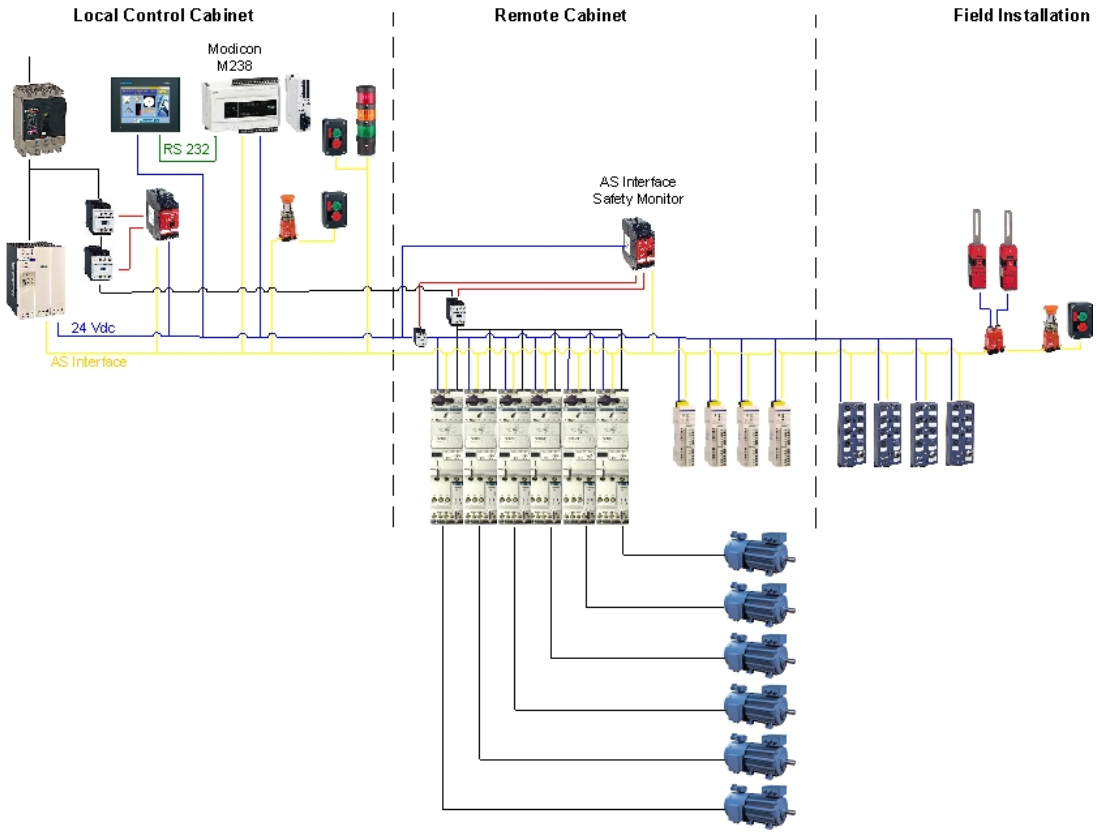
Function

Start up and functional description:

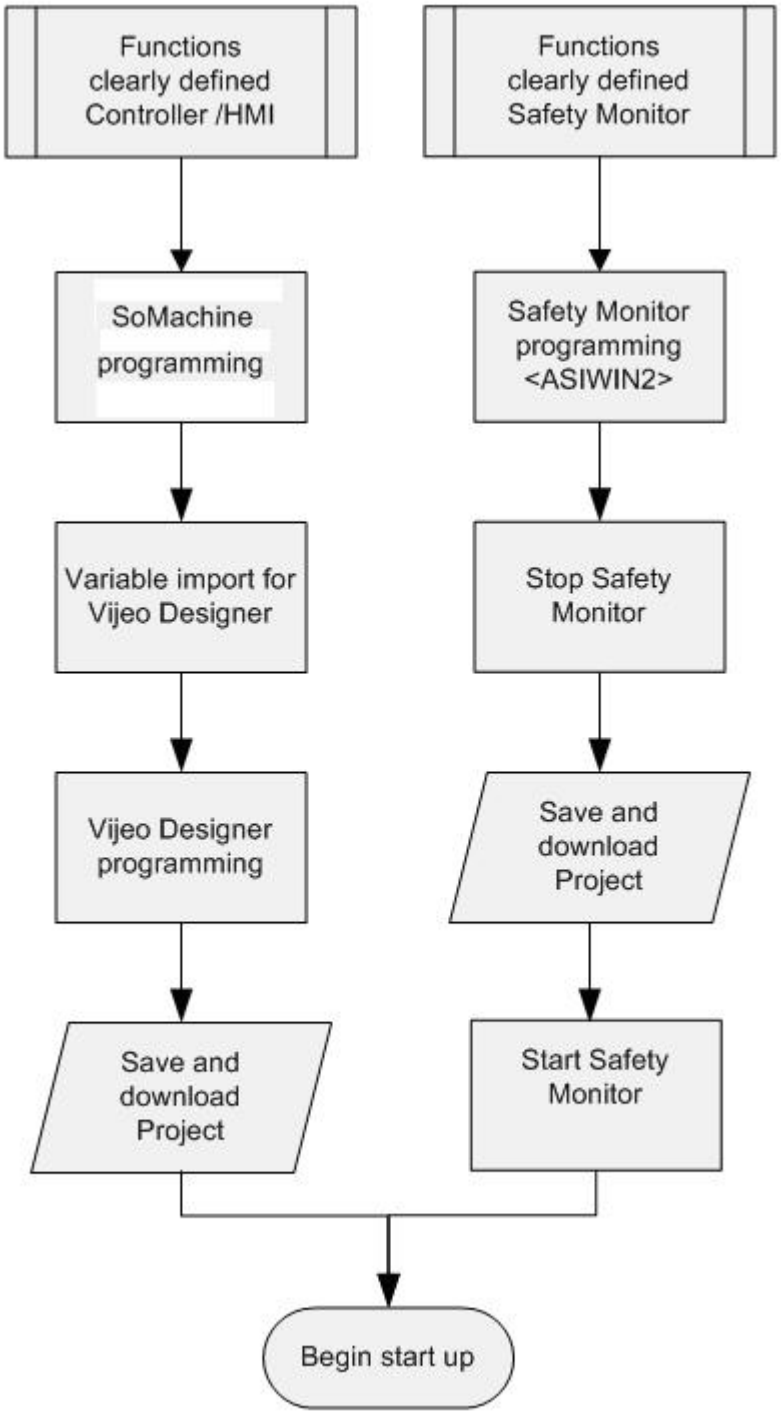
1. Switch on all fuses and contactors.
2. Switch on at main switch
3. Acknowledge the Emergency Stop
4. Check safety door(s) and acknowledge
5. Wait for the red light to turn off
6. Use the TeSys screen to control the TeSysU motor starters
7. Use the BUS, ALARM and “SAFETY” screens to control error messaging and e-stop
8. The FTB and ASI-I/O screen can be used to observe the data status of the FTBs and ASI-I/Os

Functional Layout

.



Course of Action



Communication

Introduction

This chapter describes the data passed via the communications bus that is not bound directly with digital or analog hardware.

The list contains:

- The device links
- Direction of data flow
- Symbolic name and
- Bus address of the device concerned.

Device Links

This application uses SoMachine protocol on RS485 and AS-Interface busses.

The SoMachine protocol connects:
Magelis-Panel XBT-GT
Modicon M238 Logic controller

AS-Interface connects the following devices:
1 M238 on bus address 127 (fixed)
6 TeSysU with ASILUFC5, bus addresses 09 A - 14 A
4 IP20 Input/ Output Modules, bus addresses 15 A - 18 A
4 IP 67 Input/ Output Modules, bus addresses 19 A - 22 A
2 Safety E-Stop switches, bus addresses 02 A and 03 A
2 Door guard switches, bus address 23 A
2 Safety Monitors, bus addresses 01 A and 24 A
4 XALD Pushbuttons, 05 A-08 A

AS-Interface Slaves

	Data Exchange Controller to HMI **M238** (Modbus Slave) **-> XBTGT** (Modbus-Master)	
	Symbol	**Designation**
NOTAUS1	NOTAUS1	E-Stop button, slave address 02A, BYTE NOTAUS1 (from AS-Interface address %IB3)
NOTAUS2	NOTAUS2	E-Stop button, slave address 03A, BYTE NOTAUS2 (from AS-Interface address %IB4)
NOTAUS3	NOTAUS3	Door Guard, slave address 23A, BYTE NOTAUS3 (from AS-Interface address %IB23)
AMPEL	uiXVBCred	Harmony Indicator XVBC Slave address 04A, red (from AS-Interface address %QX3.0)
	uiXVBCyel	Harmony Indicator XVBC Slave address 04A, yellow (from AS-Interface address %QX3.1)
	uiXVBCgreen	Harmony Indicator XVBC Slave address 04A, green (from AS-Interface address %QX3.2)
Pushbutton1	uixal2003_1re	Light of Pushbutton XAL Slave address 05A, red (from AS-Interface address %QX4.1
	Application.GVL.xEStpLamp	Light of Pushbutton XAL Slave address 05A, green (from AS-Interface address %QX4.0
	uixal2003_1NO	Pushbutton XAL Slave address 05A, red (from AS-Interface address %iX6.2
	uixal2003_1NC	Pushbutton XAL Slave address 05A, red (from AS-Interface address %IX6.3

AS-Interface Slaves (contd.)	uixal2003_2re	Light of Pushbutton XAL Slave address 06A, red (from AS-Interface address %QX5.1
Pushbutton2	uixal2003_2NO	Pushbutton XAL Slave address 06A, red (from AS-Interface address %iX7.2
	uixal2003_2NC	Pushbutton XAL Slave address 06A, red (from AS-Interface address %IX7.3
Pushbutton3	uixal2003_3re	Light of Pushbutton XAL Slave address 07A, red (from AS-Interface address %QX6.1
	Application.GVL.xEstoplamp2und3	Light of Pushbutton XAL Slave address 07A, green (from AS-Interface address %QX6.0
	uixal2003_3NO	Pushbutton XAL Slave address 07A, red (from AS-Interface address %iX8.2
	uixal2003_3NC	Pushbutton XAL Slave address 07A, red (from AS-Interface address %IX8.3

TeSysU (Status data)	**Data Exchange Controller to HMI** **M238 (**Modbus Slave) **-> XBTGT** (Modbus-Master)	
	Symbol	**Designation**
TeSysU 1	uiTeSysU_1Rdy	Motor starter TeSysU, slave address 09A, ready (from AS-Interface address %IX9.0)
	uiTeSysU_1Run	Motor starter TeSysU, slave address 09A,running (from AS-Interface address %IX9.1)
TeSysU 2	uiTeSysU_2Rdy	Motor starter TeSysU, slave address 10A,ready (from AS-Interface address %IX10.0)
	uiTeSysU_2Run	Motor starter TeSysU, slave address 10A,running (from AS-Interface address %IX10.1)
TeSysU 3	uiTeSysU_3Rdy	Motor starter TeSysU, slave address 11A,ready (from AS-Interface address %IX11.0)
	uiTeSysU_3Run	Motor starter TeSysU, slave address 11A,running (from AS-Interface address %IX11.1)
TeSysU 4	uiTeSysU_4Rdy	Motor starter TeSysU, slave address 12A,ready (from AS-Interface address %IX12.0)
	uiTeSysU_4Run	Motor starter TeSysU, slave address 12A,running (from AS-Interface address %IX12.1)
TeSysU 5	uiTeSysU_5Rdy	Motor starter TeSysU, slave address 13A,ready (from AS-Interface address %IX13.0)
	uiTeSysU_5Run	Motor starter TeSysU, slave address 13A,running (from AS-Interface address %IX13.1)
TeSysU 6	uiTeSysU_6Rdy	Motor starter TeSysU, slave address 14A,ready (from AS-Interface address %IX14.0)
	uiTeSysU_6Run	Motor starter TeSysU, slave address 14A,running (from AS-Interface address %IX14.1)

TeSysU (Control data)

Data Exchange Controller from/to HMI **M238** (Modbus Slave) ↔ **XBTGT** (Modbus Master)		
	Symbol	**Designation**
TeSysU 1	uiTeSysU_1Fwd	Motor starter TeSysU, slave address 09A, forward (to AS-Interface address %QX7.0)
	uiTeSysU_1Rev	Motor starter TeSysU, slave address 09A, reverse (to AS-Interface address %QX7.1)
TeSysU 2	uiTeSysU_2Fwd	Motor starter TeSysU, slave address 10A, forwards (to AS-Interface address %QX8.0)
	uiTeSysU_2Rev	Motor starter TeSysU, slave address 10A, reverse (to AS-Interface address %QX8.1)
TeSysU 3	uiTeSysU_3Fwd	Motor starter TeSysU, slave address 11A, forwards (to AS-Interface address %QX9.0)
	uiTeSysU_3Rev	Motor starter TeSysU, slave address 11A, reverse (to AS-Interface address %QX9.1)
TeSysU 4	uiTeSysU_4Fwd	Motor starter TeSysU, slave address 12A, forwards (to AS-Interface address %QX10.0)
	uiTeSysU_4Rev	Motor starter TeSysU, slave address 12A, reverse (to AS-Interface address %QX10.1)
TeSysU 5	uiTeSysU_5Fwd	Motor starter TeSysU, slave address 13A, forwards (to AS-Interface address %QX11.0)
	uiTeSysU_5Rev	Motor starter TeSysU, slave address 13A, reverse (to AS-Interface address %QX11.1)
TeSysU 6	uiTeSysU_6Fwd	Motor starter TeSysU, slave address 14A, forwards (to AS-Interface address %QX12.0)
	uiTeSysU_6Rev	Motor starter TeSysU, slave address 14A, reverse (to AS-Interface address %QX12.1)

IP20 I/O modules (control data)

Data direction Controller from/to HMI **M238** (Modbus Slave) ↔ **XBTGT** (Modbus Master)		
	Symbol	**Designation**
Module 1	IP20_Input_1	IP20 module ASI20MT, slave address 15A, (to AS-Interface address %IB15)
	IP20_Output_1	IP20 module ASI20MT, slave address 15A, (to AS-Interface address %QB15)
Module 2	IP20_Input_2	IP20 module ASI20MT, slave address 16A, (to AS-Interface address %IB16)
	IP20_Output_2	IP20 module ASI20MT, slave address 16A, (to AS-Interface address %QB16)
Module 3	IP20_Input_3	IP20 module ASI20MT, slave address 17A, (to AS-Interface address %IB17)
	IP20_Output_3	IP20 module ASI20MT, slave address 17A, (to AS-Interface address %QB17)
Module 4	IP20_Input_4	IP20 module ASI20MT, slave address 18A, (to AS-Interface address %IB18)
	IP20_Output_4	IP20 module ASI20MT, slave address 18A, (to AS-Interface address %QB18)

IP67 I/O modules (control data)	**Data direction Controller from/to HMI** **M238** (Modbus Slave) **<-> XBTGT** (Modbus Master)	
	Symbol	**Designation**
Module 1	IP67_Input_1	IP67 module ASI67FMP, slave address 19A, (to AS-Interface address %IB15)
	IP67_Output_1	IP67 module ASI67FMP, slave address 19A, (to AS-Interface address %QB15)
Module 2	IP67_Input_2	IP67 module ASI67FMP, slave address 20A, (to AS-Interface address %IB16)
	IP67_Output_2	IP67 module ASI67FMP, slave address 20A, (to AS-Interface address %QB16)
Module 3	IP67_Input_3	IP67 module ASI67FMP, slave address 21A, (to AS-Interface address %IB17)
	IP67_Output_3	IP67 module ASI67FMP, slave address 21A, (to AS-Interface address %QB17)
Module 4	IP67_Input_4	IP67 module ASI67FMP, slave address 22A, (to AS-Interface address %IB18)
	IP67_Output_4	IP67 module ASI67FMP, slave address 22A, (to AS-Interface address %QB18)

Controller

Introduction

The controller chapter describes the steps required for the initialization and configuration and the source program required to fulfill the functions.

Requirements

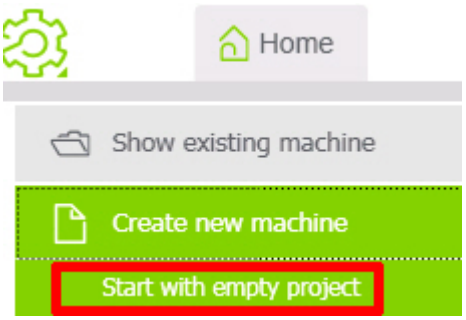
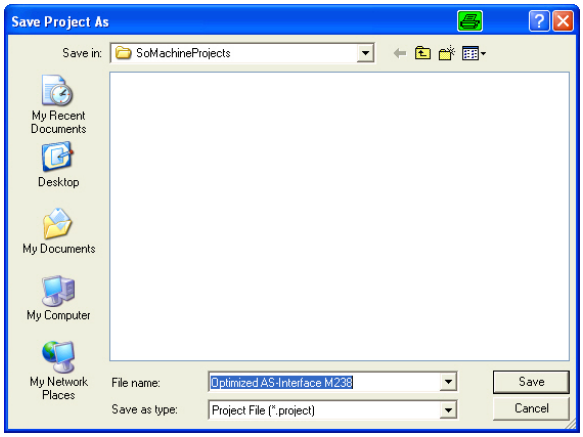
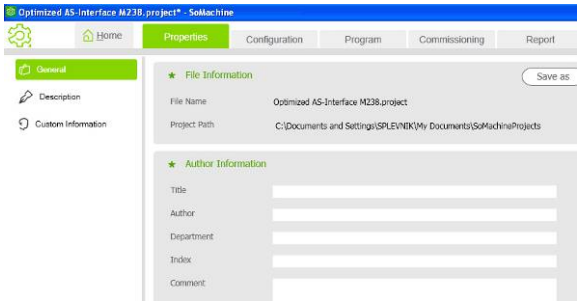
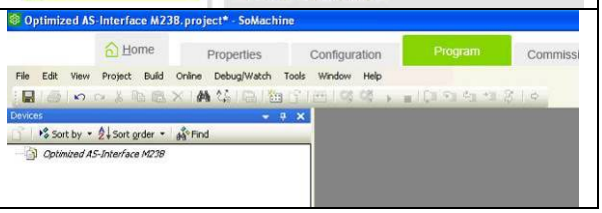
The following is required before proceeding with the controller configuration:

- SoMachine software tool is installed on your PC
- The Modicon M238 Logic controller is switched on and running
- The controller is connected to the HMI with the XBTZ9008 communication cable (controller to HMI)
- The HMI is connected to the PC via the cable XBTGZ935 (HMI to PC)

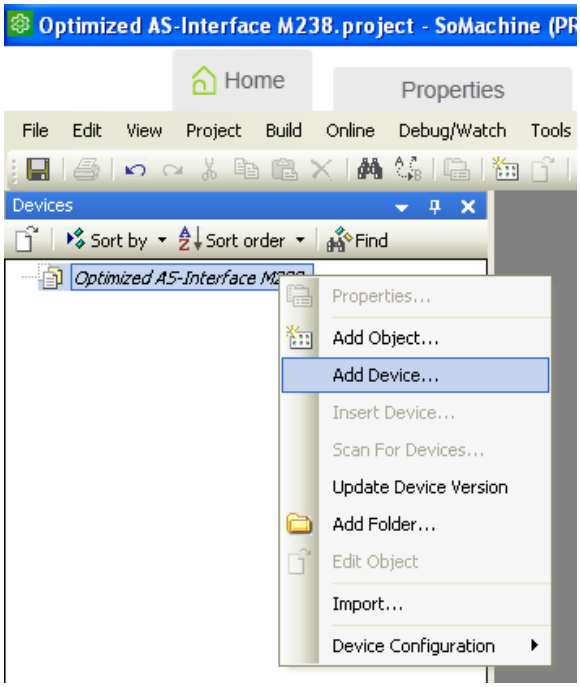
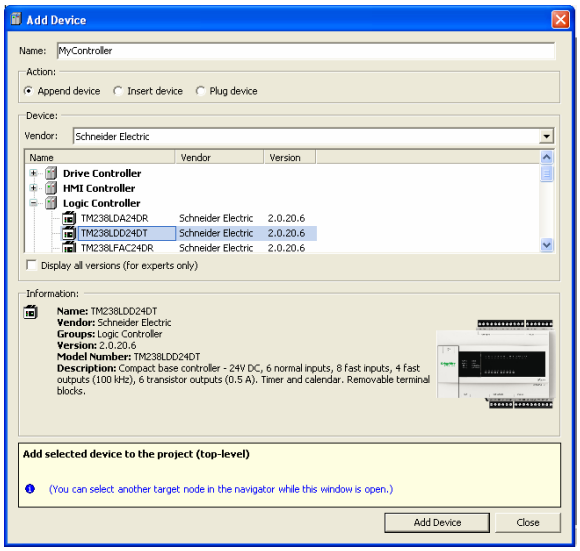
Setting up the controller is done as follows:

- Create a new project
- Add the Controller
- Add an AS-Interface extension module
- Add AS-Interface Devices
- Set the Parameters and Addresses
- Creating Variables
- Add a POU
- Task configuration
- Add Toolbox library
- Configure controller ↔ HMI data exchange
- Add Vijeo Designer HMI
- Communication setting controller ↔ PC
- Communication setting controller ↔ HMI
- Save the Project
- Build application
- Download the controller and HMI program
- Login to the controller
- Application overview

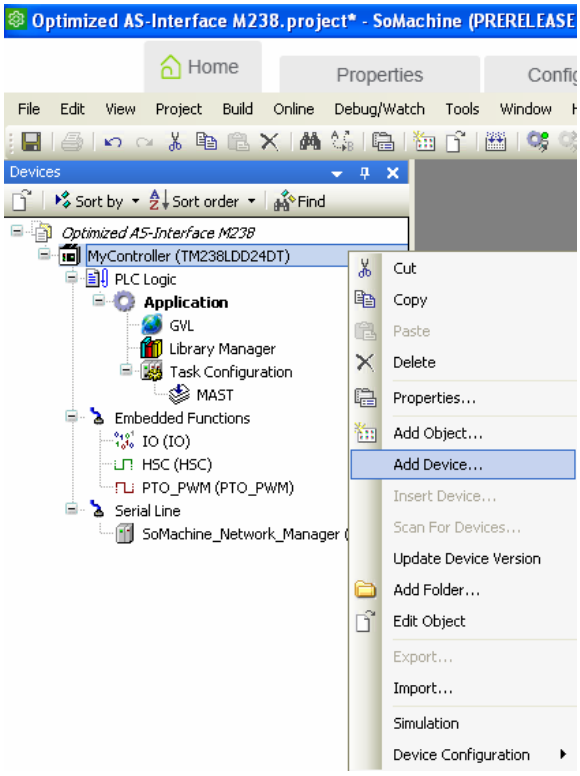
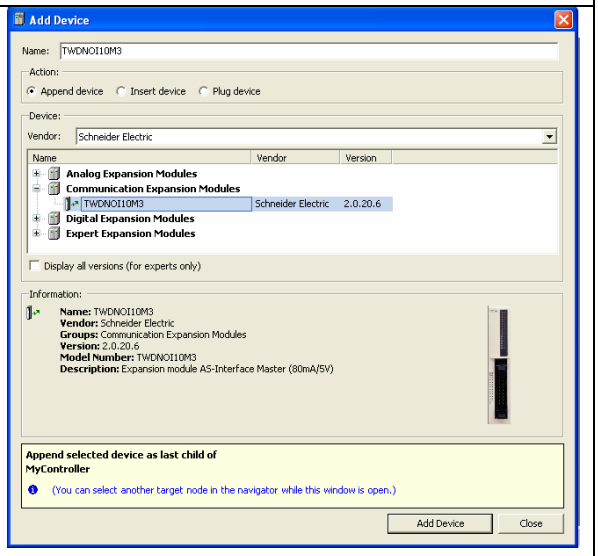
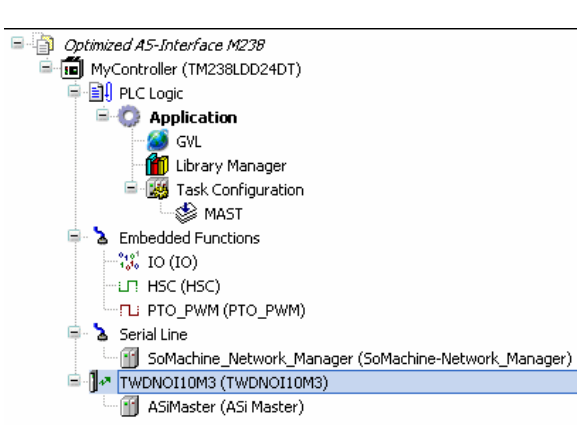
Create a new project

1	To create a new project select: **Create new machine→** **Start with empty project**	
2	In the **Save Project As** dialog enter a **File name** and select the location. Click on **Save.** **NOTE:** By default the project is saved under My Documents.	
3	The SoMachine desktop opens.	
4	Select the **Program** tab	
5	The Program window appears.	

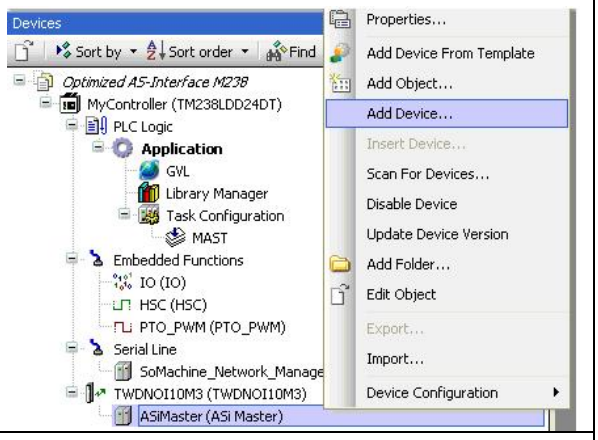
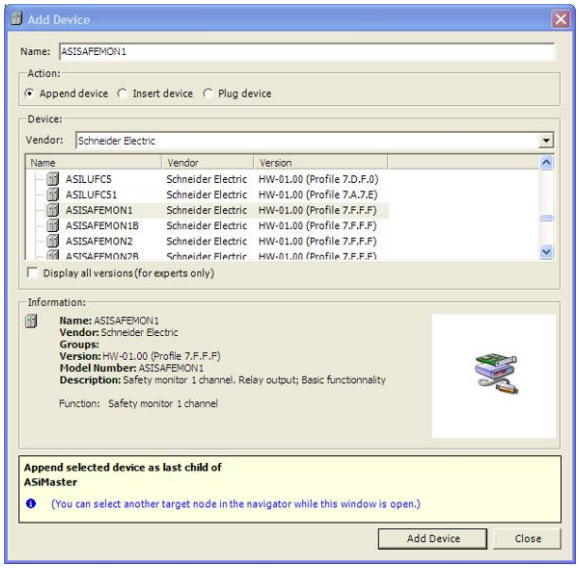
Add Device

1	Right click on the name of your program in the browser and in the pop-up menu click on **Add Device…**	
	In the **Add Device** dialog select **TM238LDD24DT.** Click on the **Add Device** button.	

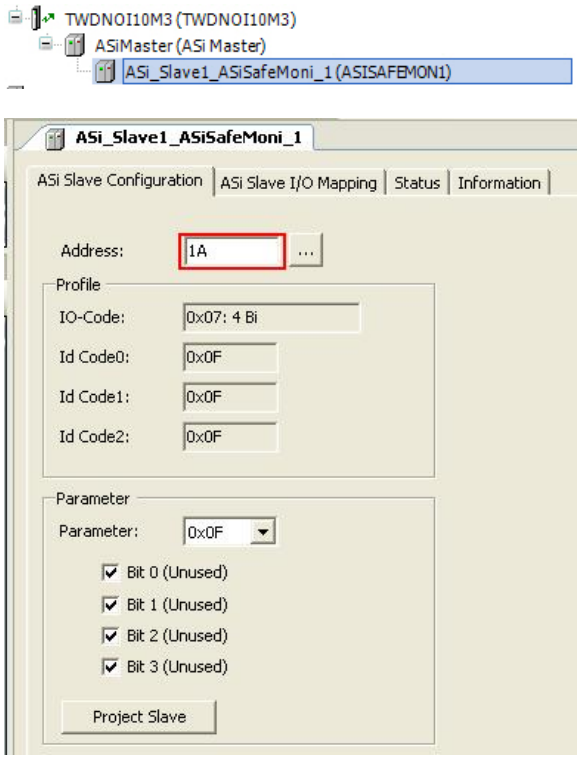
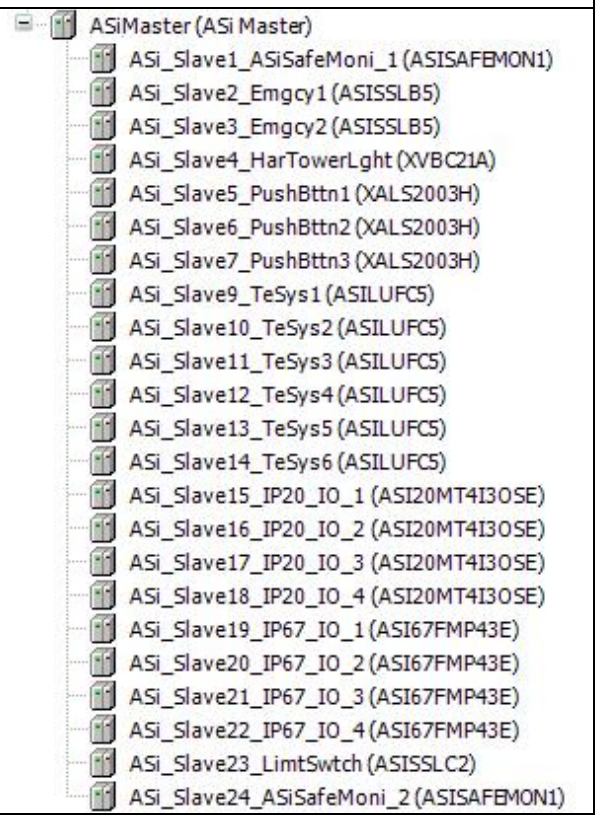
Add an AS-Interface Expansion module

1	Right click on the **MyController(TM238LDD24DT)** in the **Devices** browser. Click on **AddDevice…**	
2	Select the **TWDNOI10M3** in the **Add Device** window and click **Add Device** button	
3	The new **TWDNOI10M3** expansion module and the **ASiMaster** are now created and an entry added in the Devices browser.	

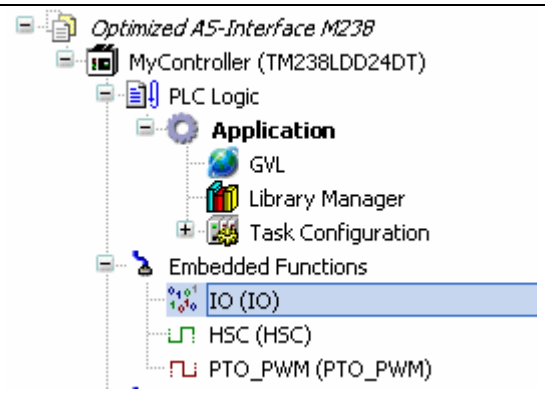
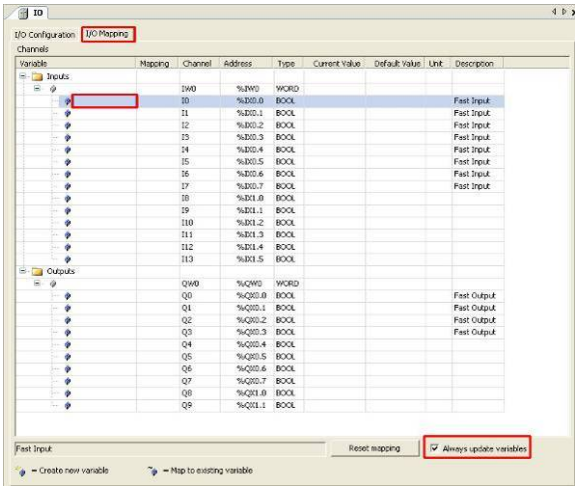
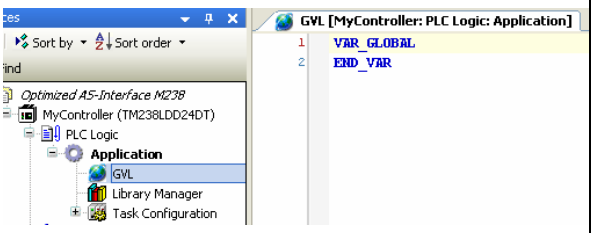
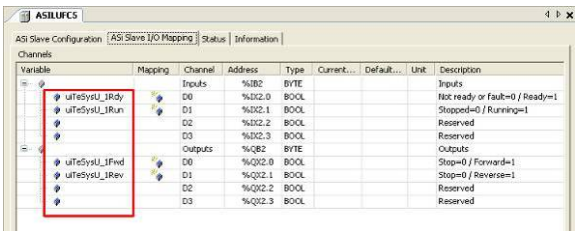
Add AS-Interface devices

1	Right click on ASiMaster in the **devices** browser and then click **Add Device…**	
2	Click on the slave and click on **Add Device** Add all the slaves you require. In our example we added: **1x ASISAFEMON1** **2x ASISSLB5** **1x XVBC21A** **3x XALS2003H** **6x ASILUFC5** **4x ASI204I3OSE** **4x ASI67FMP43E** **1x ASISSLC2** **1x ASISAFEMON1** Add the slave devices in the order mentioned above. Close the dialog after adding all the devices. **NOTE:** The name of the device can be changed in the **Name** field.	

Set the parameters and Addresses

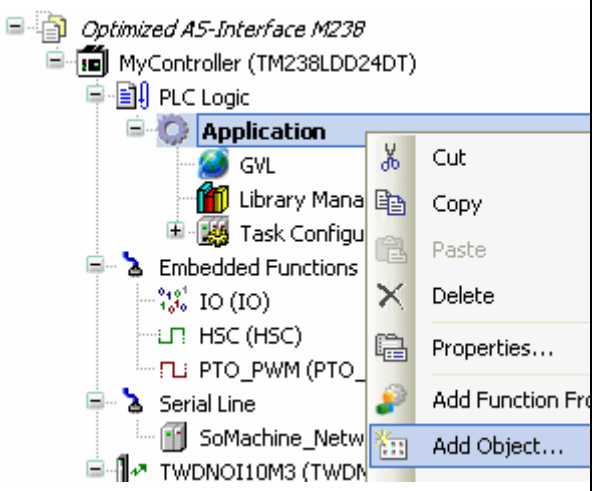
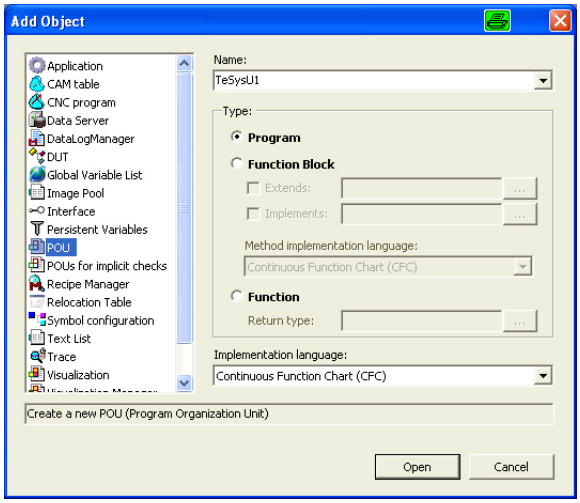
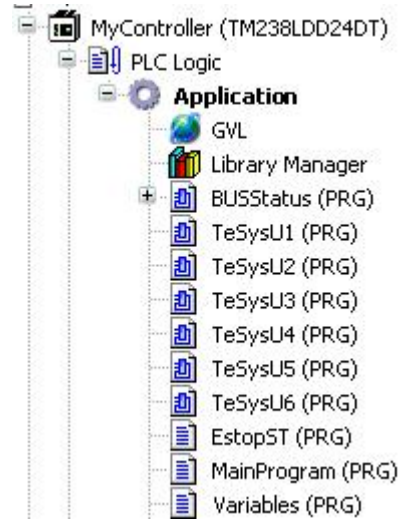
1	The devices are now listed in the browser under the **ASiMaster.** Double click on the first slave **ASi_Slave1_ASiSafeMoni_1** and set the address. In our example **1A** **NOTE:** You don't have to provide the address if you add the slaves in succession. The addresses are provided in order in which you add the slaves. The first slave you add gets 1A, the second 2A etc .	
2	List of AS-Interface node names and addresses	

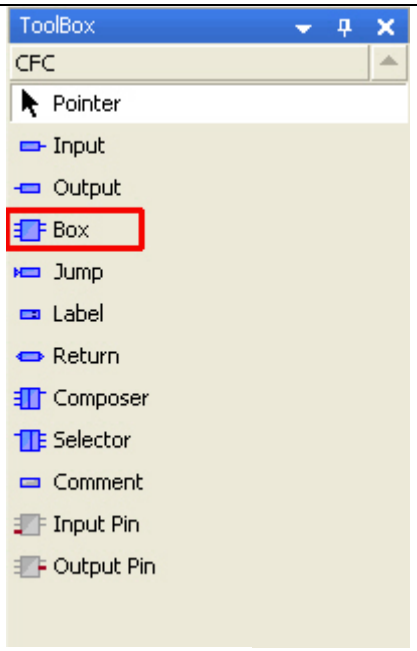
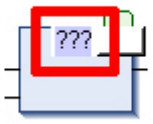
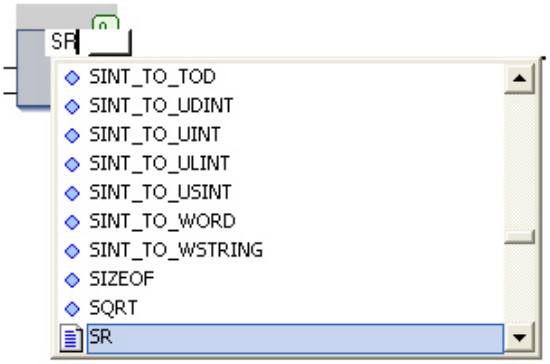
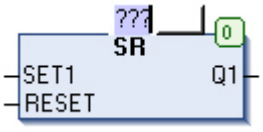
Creating Variables

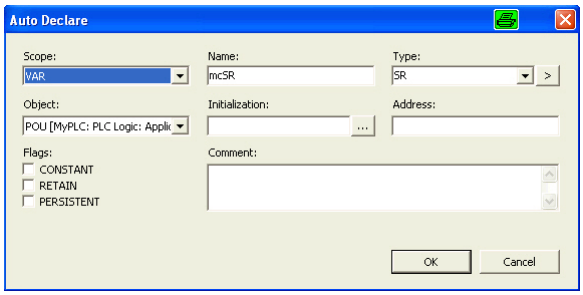
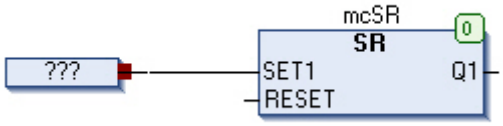
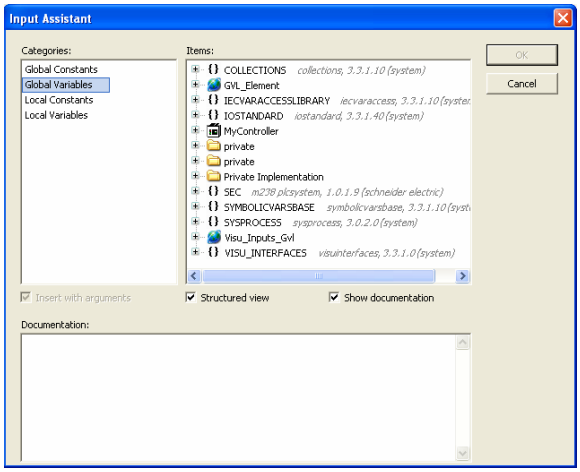
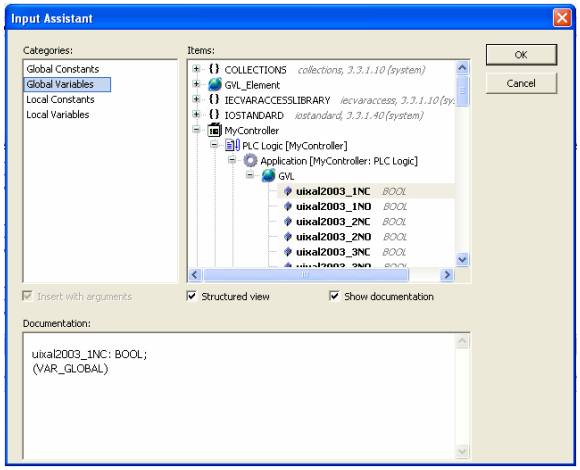
1	There are two methods of **Mapping:** 1.Mapping to an existing variable Double click on: **Embedded Functions -> IO >I/O Mapping** tab The names of the variables can be entered in the **Variable** field. To update the variables with the latest I/O data check **Always update variables.**	 
	2.Create a new variable with double click on **GVL** Every Variable which is created here can be used throughout the whole program of SoMachine	
2	Double click on an **ASILUFC5** module Click on the tab: **ASi Slave I/O Mapping** Create the following variables as in the image: **uiTesysU_1Rdy** **uiTesysU_1Run** **uiTesysU_1Fwd** **uiTesysU_1Rev** This is a typical mapping for a TeSysU. Do the same for the other ASILUFC5 Modules.	

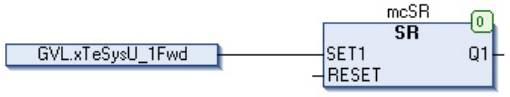
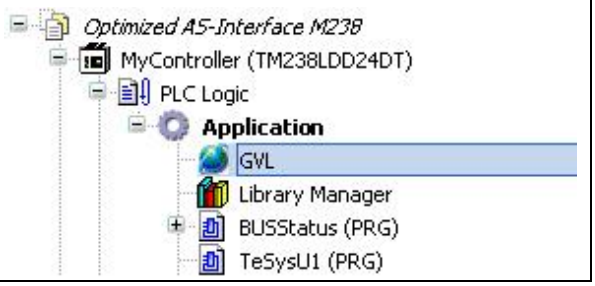
3	If you want to map more Variables on other slaves, the procedure is the same as with the described slaves.	

Add a POU

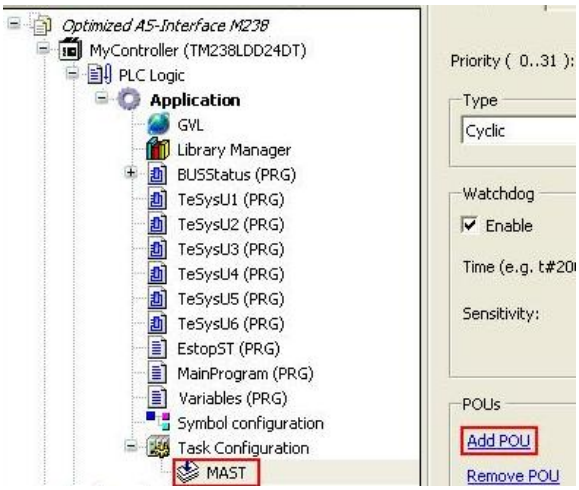
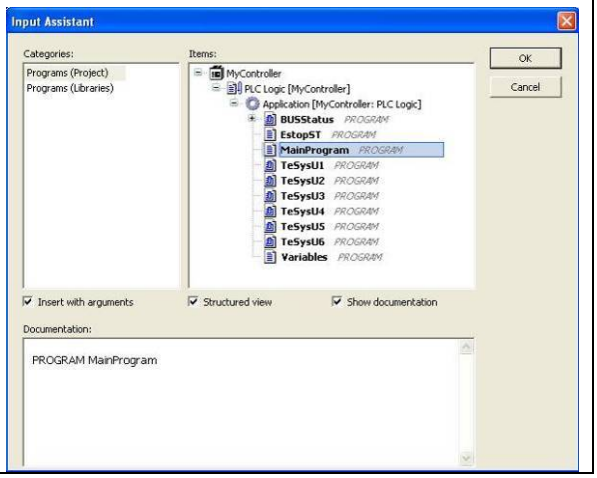
1	Right click on **Application->Add Object...**	
2	Select **POU** and enter a **Name.** In **Type** select **Program** and in **Implementation language** select **Continuous Function Chart (CFC).** It is possible to select all the IEC languages and to generate functions and function blocks. Click on **Open.**	
3	The new POU **TeSysU1** is now visible under **Application.** Similarly more POUs can be added. Double click on **TeSysU1** to open it.	

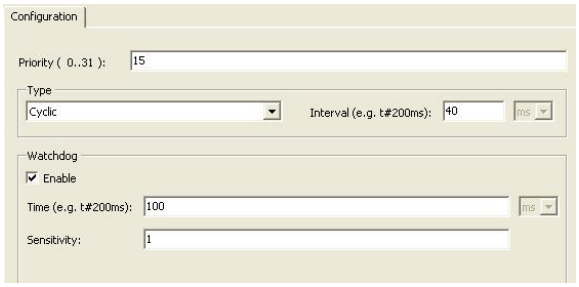
4	The upper frame displays the declaration section. The lower frame is for programming. On the right side is the **ToolBox** window**.** Use drag and drop with the Toolbox to place example templates in the programming section.	
5	Select the **Box** to add it in the POU. When you have placed a template in the programming section click on **???**	 
6	Type in the name for the function or function block. As the first letters are typed in a list box opens up with hints for the name. In this project example an SR FB was selected.	
7	To instantiate the FB click on **???**…	

8	…and type in the name (for example mcSR). Now press Enter. The **Auto Declare** dialog opens. If you wish to add a comment you can do this in the **Comment** box**.** Click on **OK** to create the instance.	
9	To connect a variable to an input place on the input side of the FB, connect the input box to the FB input by clicking on the red field and dragging it to the input of the FB.	
10	Click the input field and press **F8.** The **Input Assistant** is displayed.	
11	In the **Input Assistant** select **Global Variables** in the **Categories** list**.** Then select: **MyController→ PLC Logic →Application[MyController: PLC Logic] → GVL** and then the variable. Click on **OK.**	

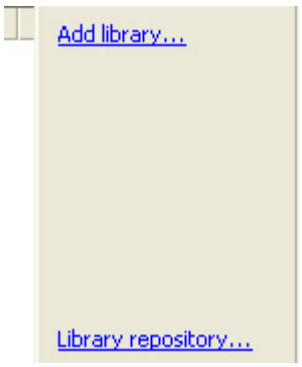
12	The Input is now displayed in the input box.	
13	The **VAR_GLOBAL** variables are located in the **GVL** folder. All variables located in this folder can be accessed throughout the whole **Application**. If the variables are located in the POU, they can only be accessed by the POU (local variables).	

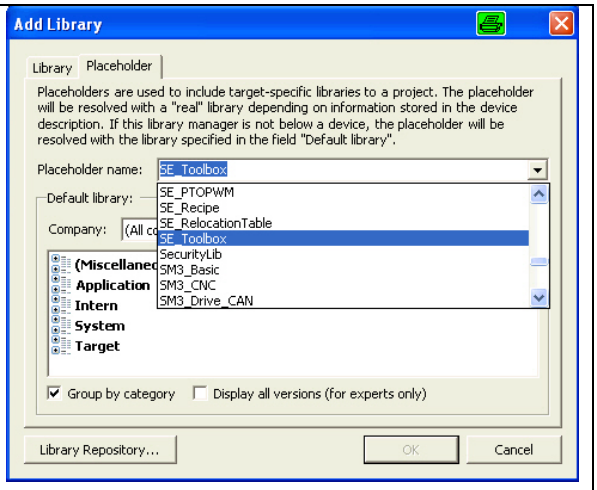
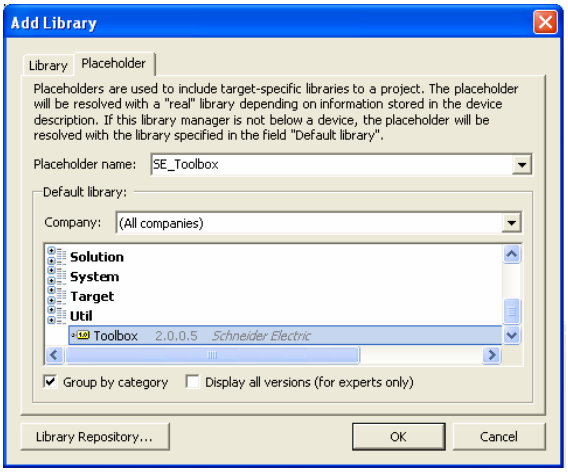
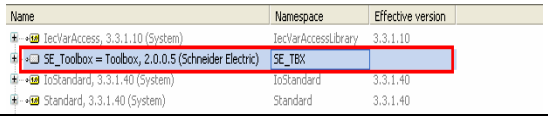
Task configuration

1	Before you can start working with the new POU you have to add it to a **Task**. Here, the POUs are added to the **MAST** task. To do this double click the **MAST** task and click on **Add POU.**	
2	Select **Programs (Project)** in the **Categories** list and select the **MainProgram** in the **Items** list**.** Then click **OK**.	

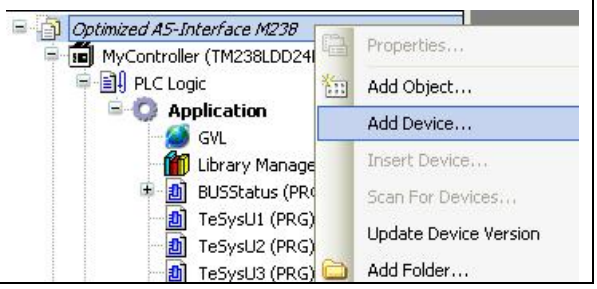
3	Now the POU is in the **MAST** task. In the upper part of the **MAST** task configuration you can change the **Type** of the task. In this project it is **Cyclic.** Directly under the **Type** menu is the **Watchdog** field. Set the Watchdog **time** to **100** ms.	

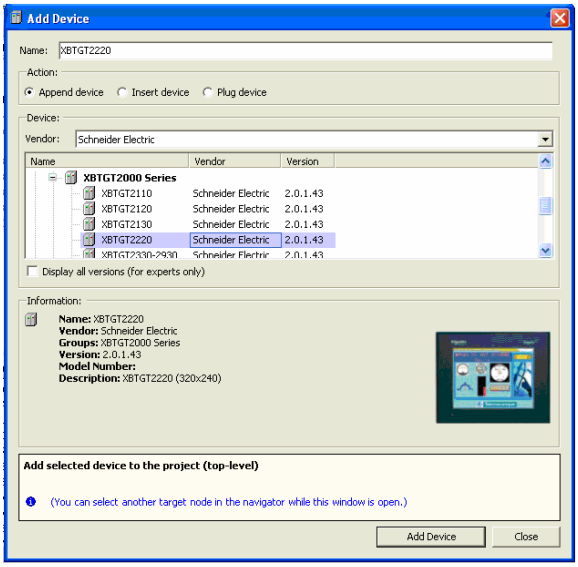
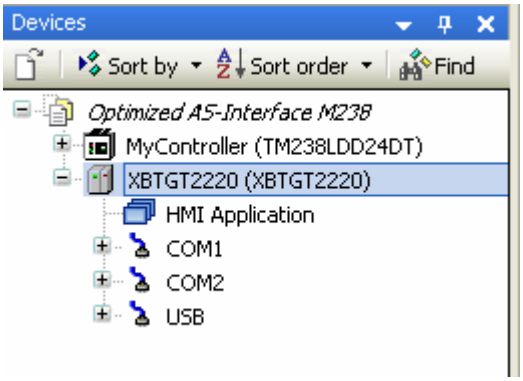
Add Library

1	To use some special functions you need special libraries. These can be inserted by double clicking on the **Library Manager.**	
2	In the Library Manager click on: **Add library…**	

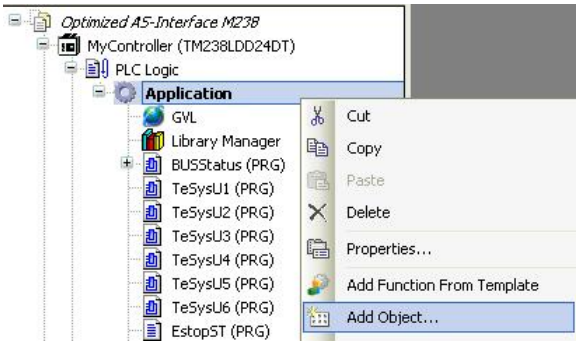
3	Example: In the **Add Library --> Placeholder** dialog select: **Placeholder name->** **SE_Toolbox** Select **Util -> Toolbox** and click on **OK**	 
4	Now the new libraries can be seen in the Library Manager.	
5	Steps 1 to 3 have to be executed in case other user libraries have to be included	

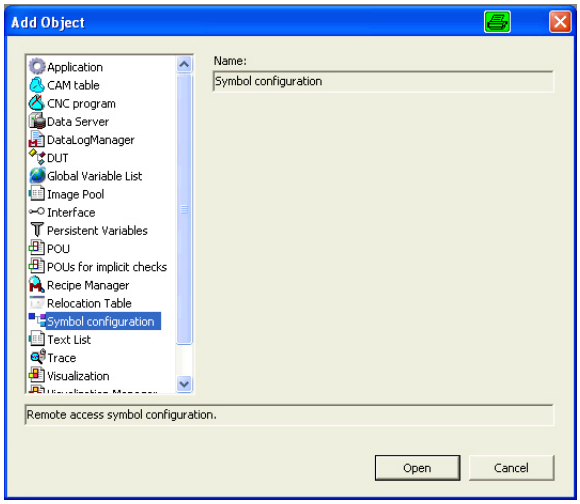
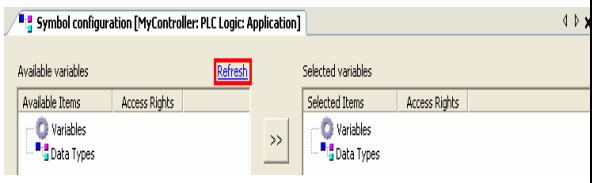
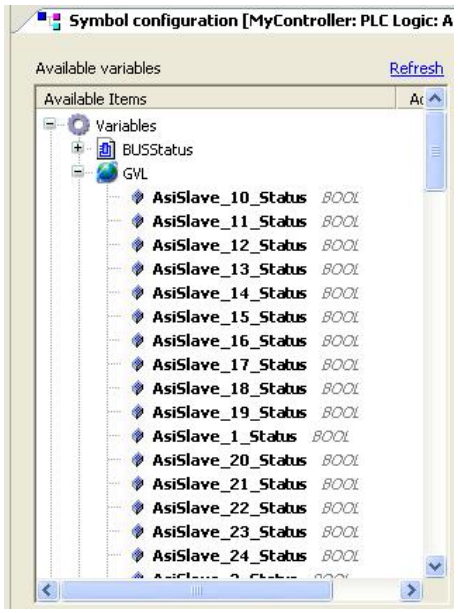
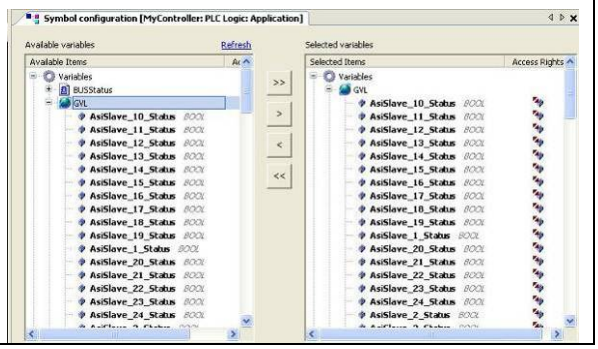
Add Vijeo Designer HMI

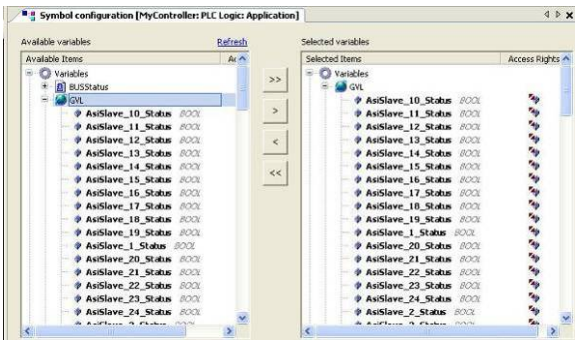
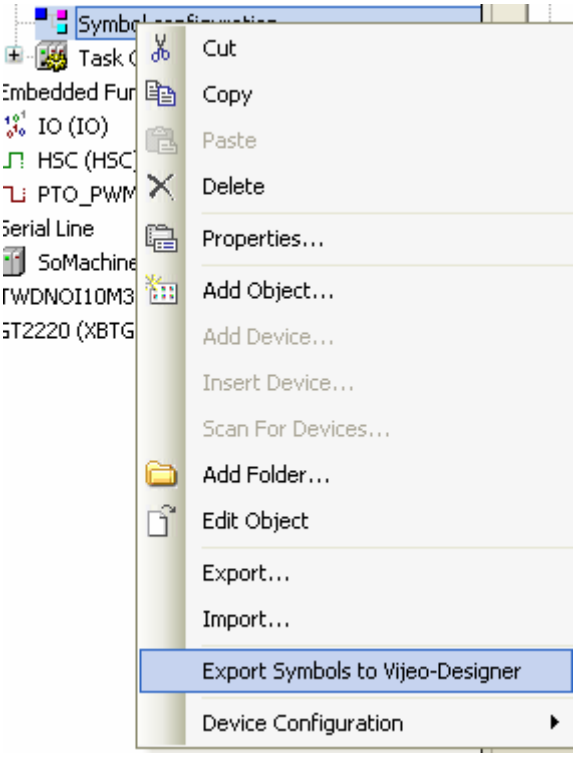
1	To add a Vijeo Designer HMI to the project right click on: **Optimized AS-Interface M238->Add Device…**	

2	In the **Add Device** dialog select **Device** and select **XBTGT2220** and then click on **Add Device**	
3	The new **XBTGT2220** is now listed in the project browser. **NOTE:** With this **XBTGT2220,** the Program Vijeo Designer opens, and you can start your programming. (See chapter HMI)	

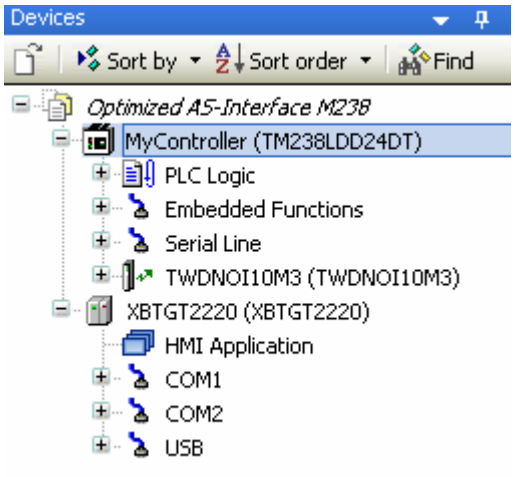
Configure controller ↔ HMI data exchange

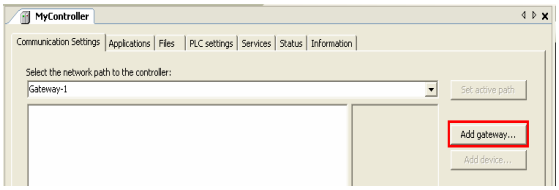
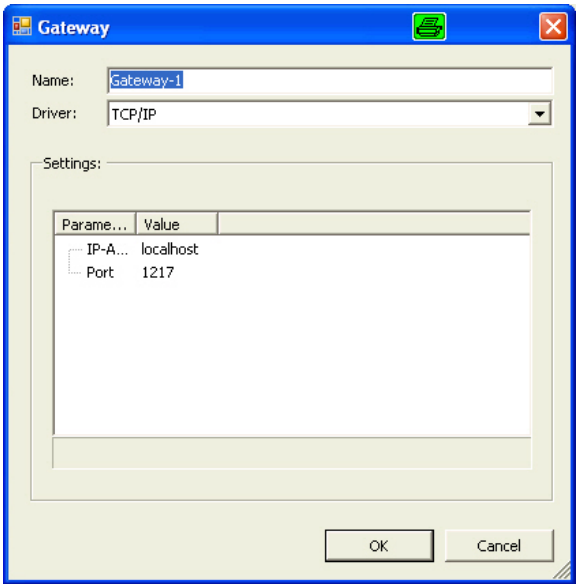
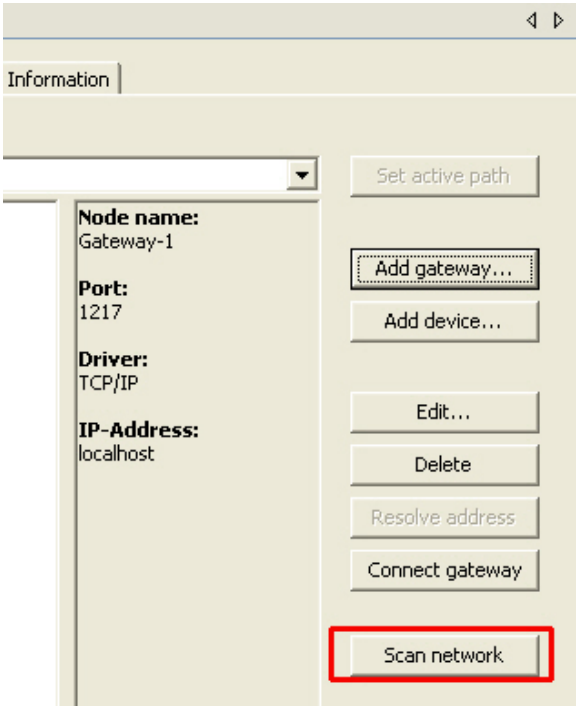
1	Right click on: **Application->Add Object...**	

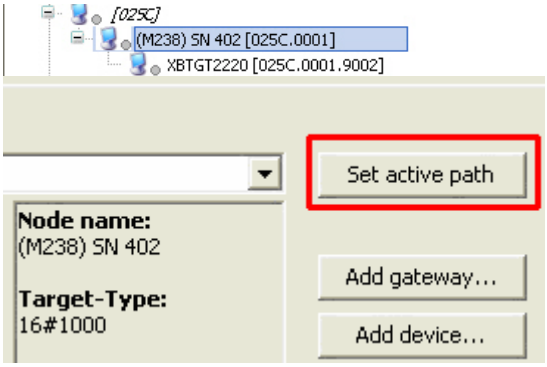
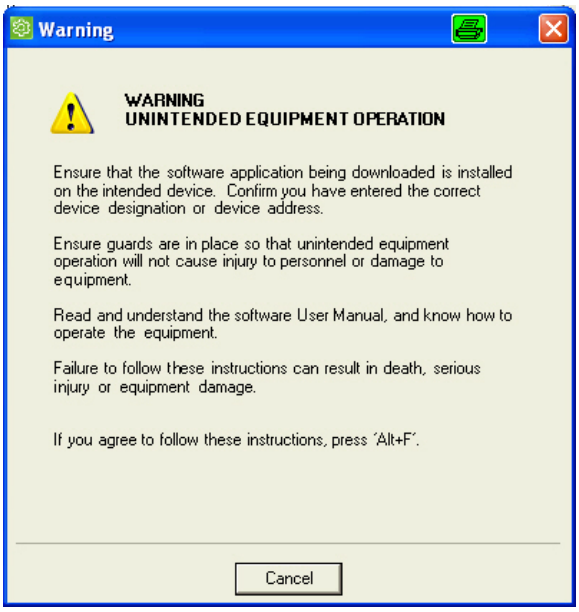
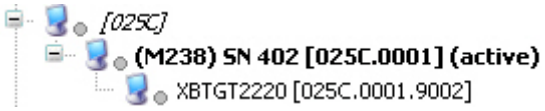
2	In the **Add Object** dialog. Select **Symbol configuration** Click on **Open**.	
3	Click on **Refresh** in the now open Symbol configuration.	
4	All Variables created in the user program are shown in the variables list. In this project all variables are global variables and, as such, are located in the **GVL** folder. To export variables to the HMI, select them and click on **>**.	 

5	The right frame lists the **Selected Variables** which are to be used in the HMI.	
6	Right click on **Symbol configuration -> Export Symbols to Vijeo-Designer** to export the variable list	

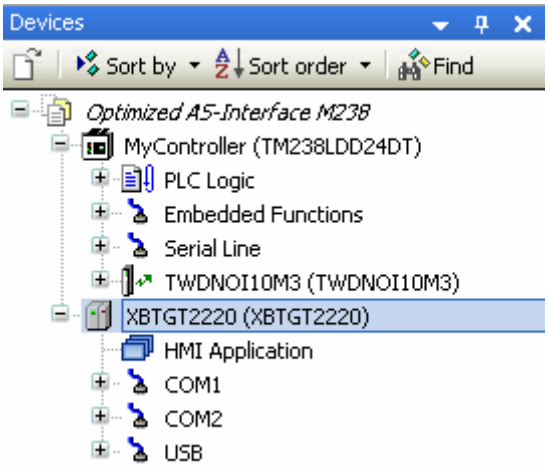
Communication setting controller ↔ PC

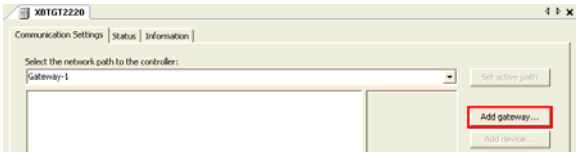
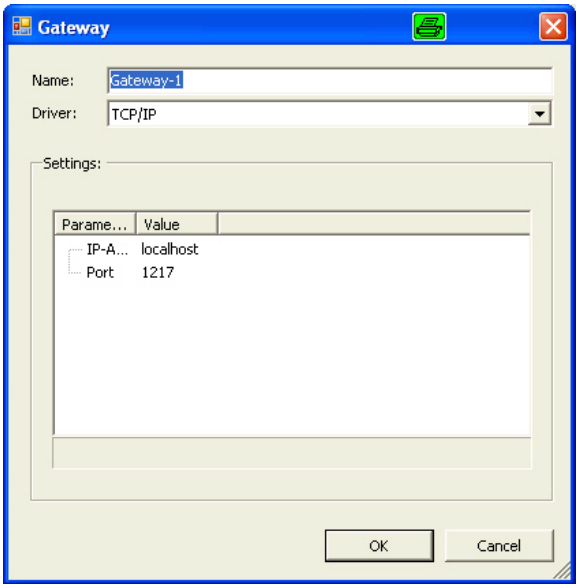
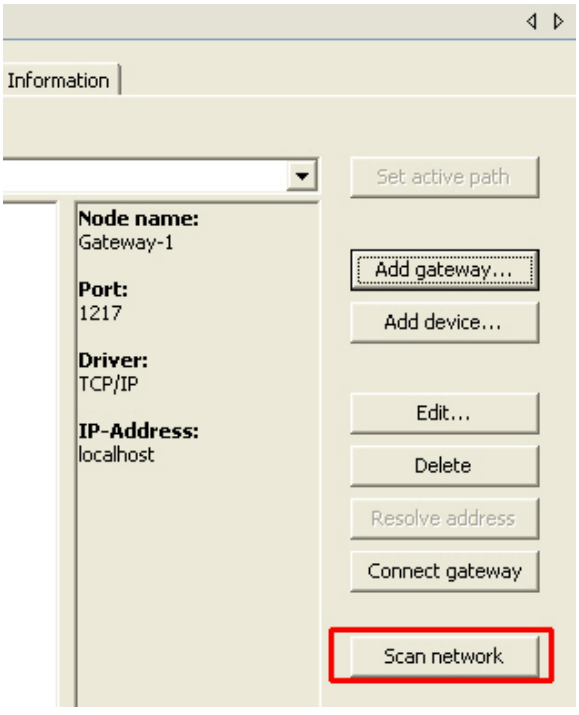
1	To configure the communication gateway double click on **MyController**	

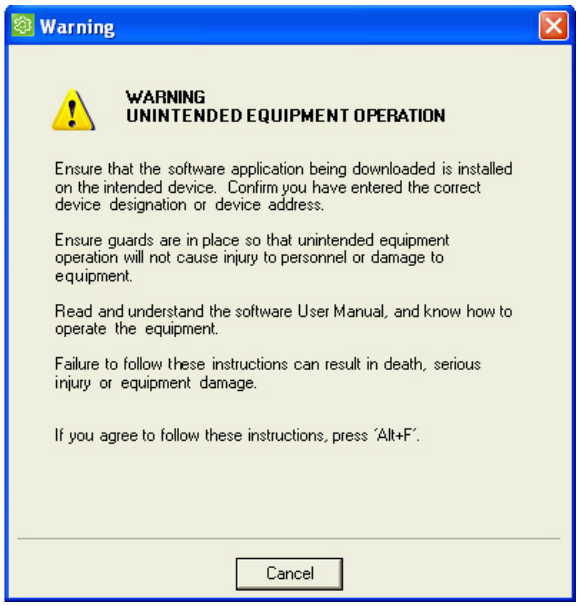
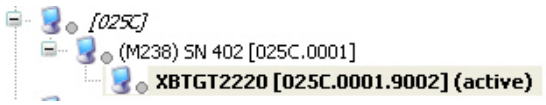
2	On the **Communication Settings** tab click on: **Add gateway...**	
3	Keep the default settings and click on **OK.**	
4	Select **Gateway-1** and click on **Scan Network.**	

5	When the scan is finished, the devices pop up under the gateway. Select the used controller and click **Set active path.**	
6	A warning pop-up window opens and the text must be read.	
7	The used controller is now marked as **active.**	

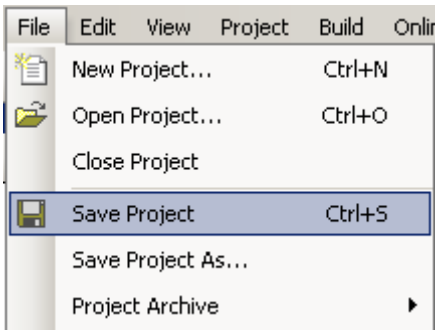
Communication setting HMI ↔ PC

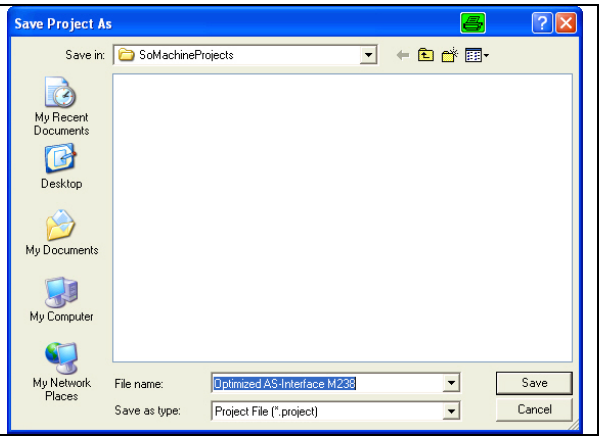
1	To configure the communication gateway double click on: **XBTGT2220**.	

2	On the **Communication Settings** tab and click on **Add gateway...**	
3	Accept the default settings by clicking on **OK.**	
4	Select **Gateway-1** and click on **Scan Network.**	

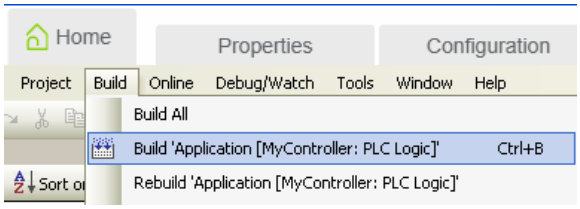
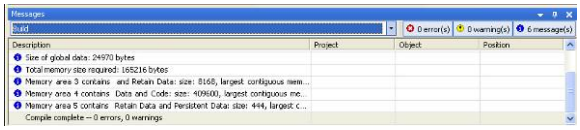
5	When the scan is finished, the devices are listed under the gateway. Select the used HMI and click on **Set active path.**	
6	A warning pop up window opens and the text must be read.	
7	The used HMI is now marked as **active.**	

Save the Project

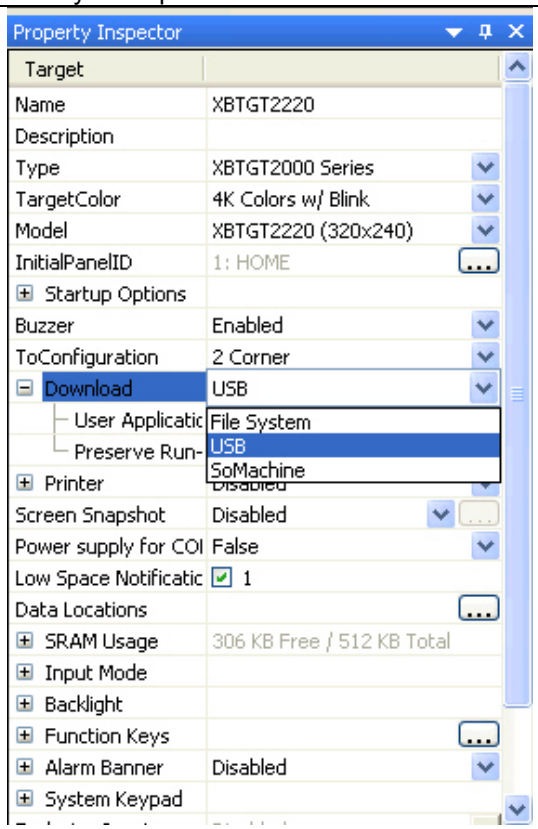
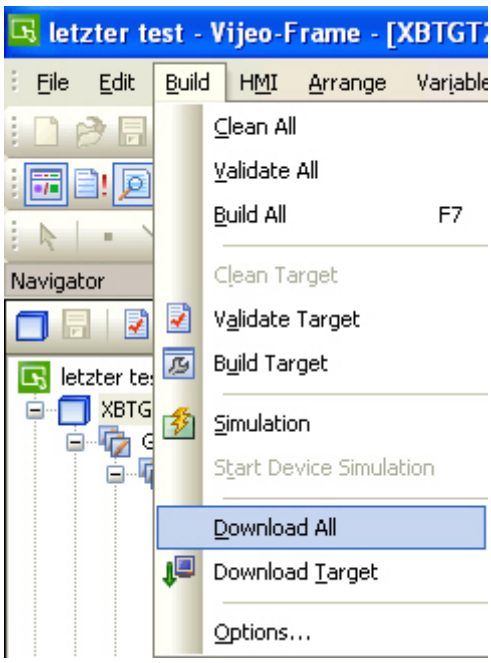
1	To save the project, click **File → Save Project** To save the project under a different name, click **File → Save Project As…**	

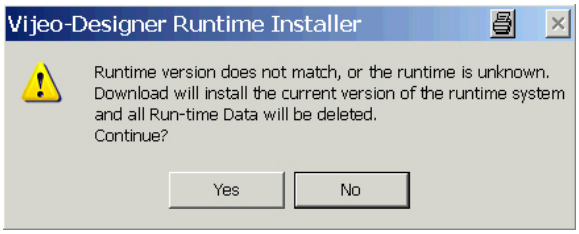
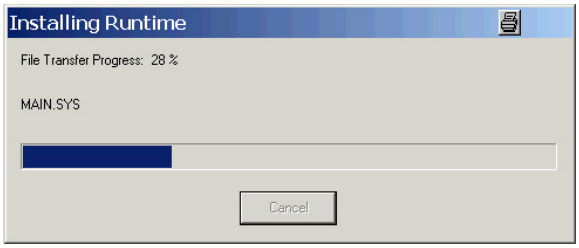
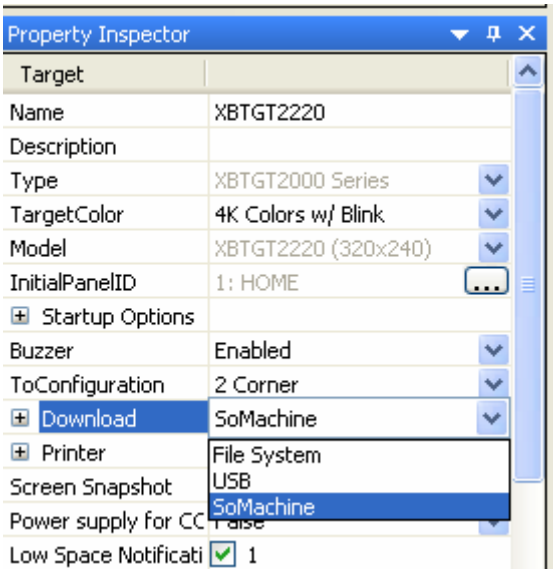
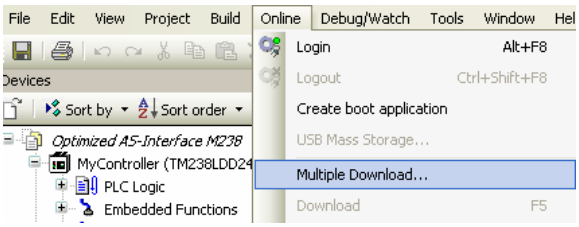
2	In the **Save Project As** dialog, enter the new **File name** and click on **Save**.	

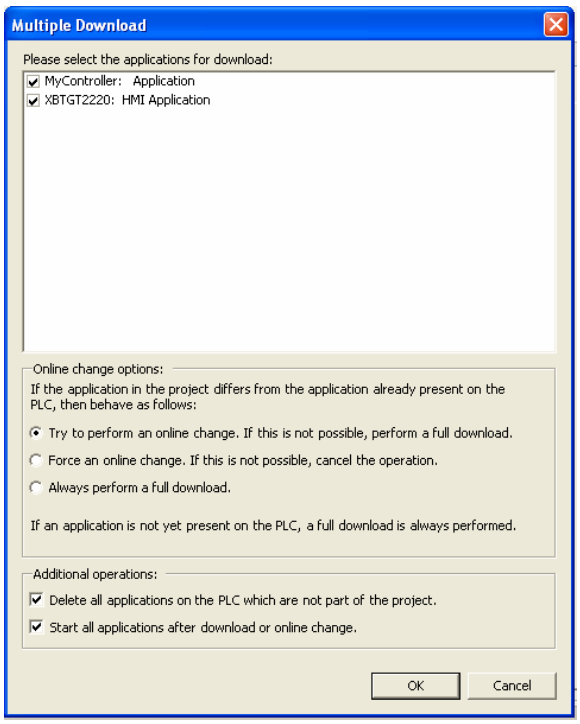
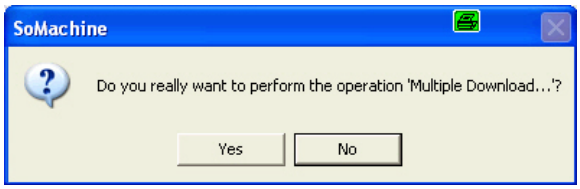
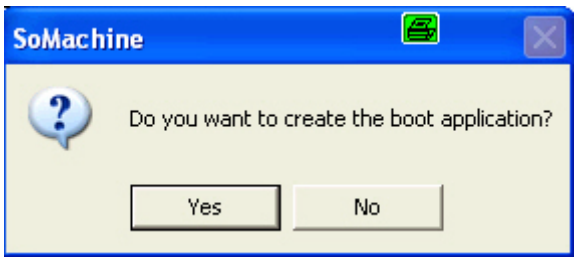
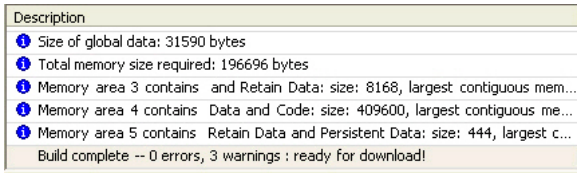
Build Application

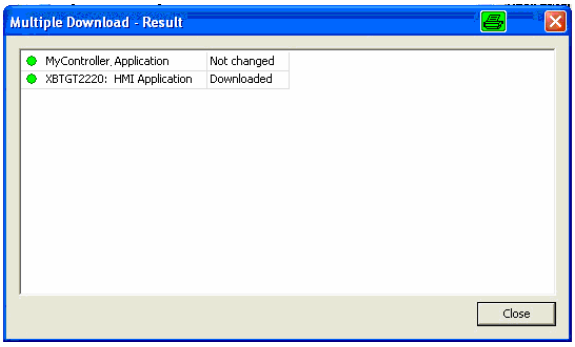
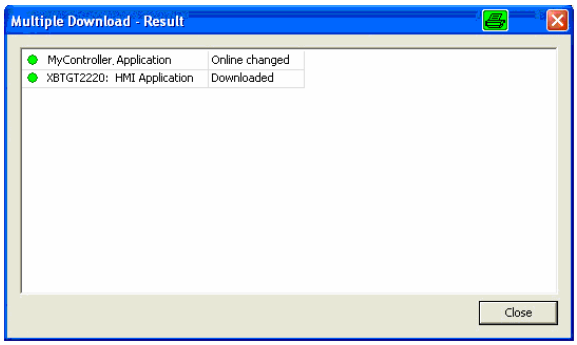
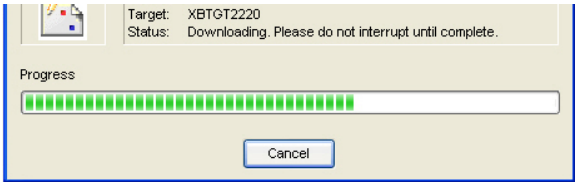
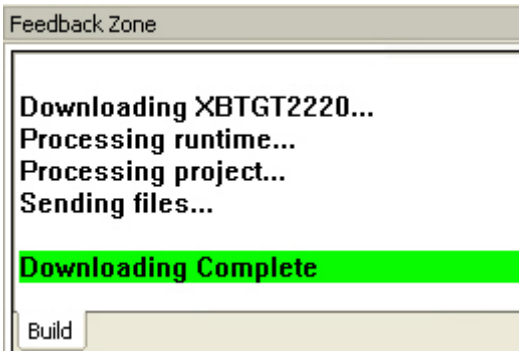
1	To build the application click on **Build→Build 'Application [MyController: PLC Logic]'.** **Note:** If you wish to build the whole project (HMI and controller) click **Build all**	
2	After the build you are notified in the **Message** field as to whether the build was successful or not. If the build was not successful there will be an alert list in the **Message** field.	

Download the controller and HMI project

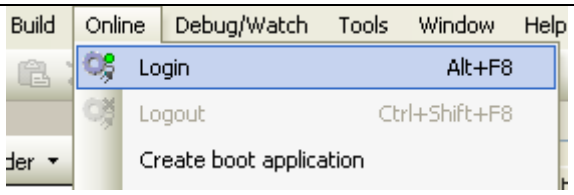
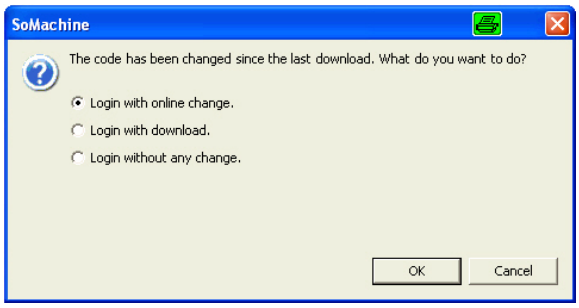
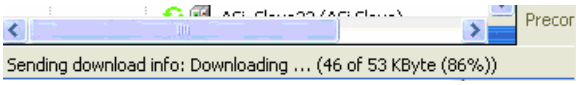
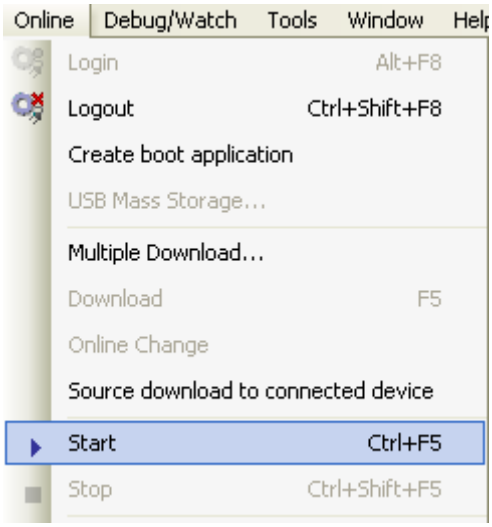
1	**Note:** If it is the first time you are connecting to the HMI you have to first download the latest runtime version to the HMI using Vijeo Designer. This first download is described in the following steps. If this is not the first download go directly to step 7	
2	In Vijeo Designer in the **Property Inspector** select **Download** via **USB.** **Note:** The PC must be connected to the HMI via the cable XBTZG935.	
3	**Vijeo Designer download**: Select: **Build->Download All**	

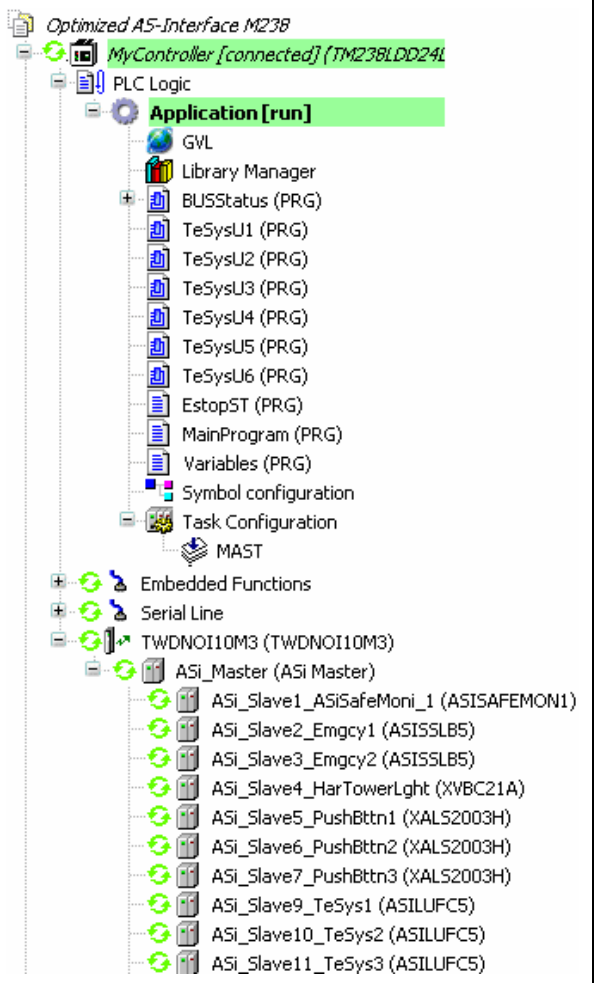
4	The **Vijeo-Designer Runtime Installer** dialog indicates that the runtime versions do not match. Start the download of the new version by clicking on **Yes**	
5	The actual state of the download is displayed in a progress bar.	
6	Once the runtime is downloaded, change the **Download** connection in the **Property Inspector** back to **SoMachine.**	
7	**SoMachine download**: To download the application to the controller and the HMI click **Online→Multiple Download…**	

8	Select the controller **MyController** and the HMI **XBTGT2220** click on **OK**.	
9	SoMachine asks you if you really want to perform the operation and if you want to create a boot application. Please confirm both with Yes Before the download starts a build of the complete project is done. The result of the build is displayed in the **Messages** window.	  

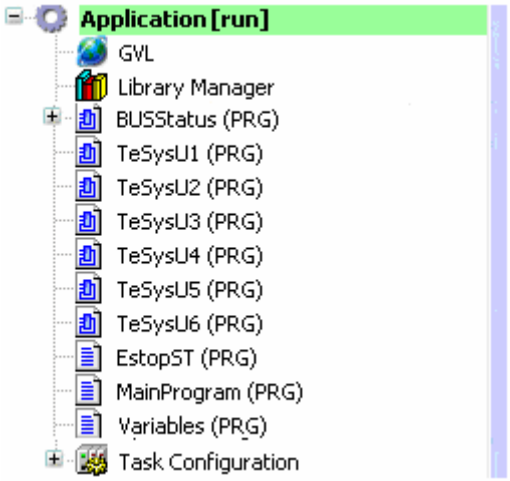
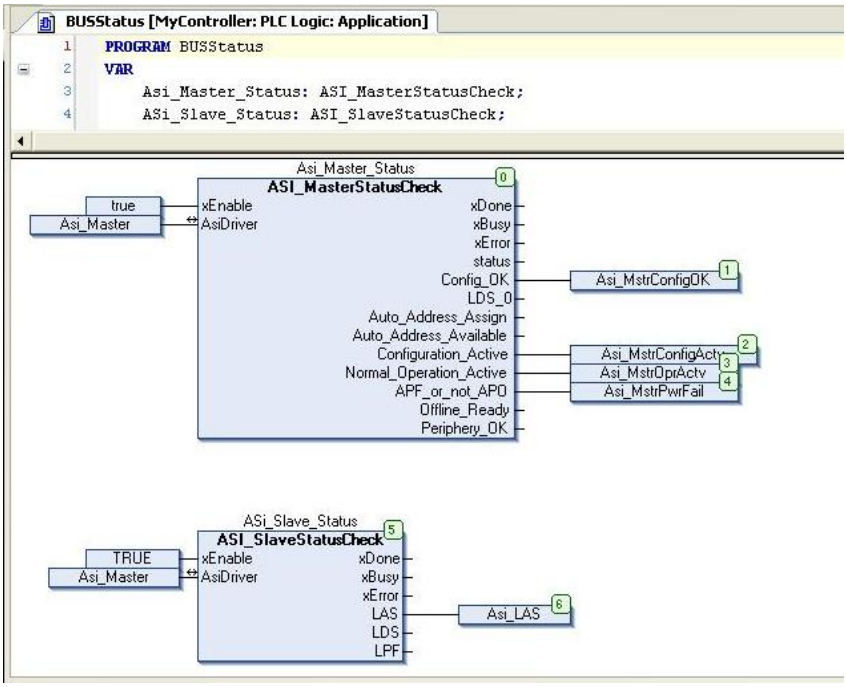
10	The results of the download to the controller are displayed in the **Multiple Download – Result** window**.** Here are two examples: In the first dialog there was no change. And in the second dialog there was an online change done. Click on **Close** to close to the results window**.**	 
11	Once the download to the controller is finished, the HMI download starts	
12	The result of the HMI download is displayed in the **Message** window.	

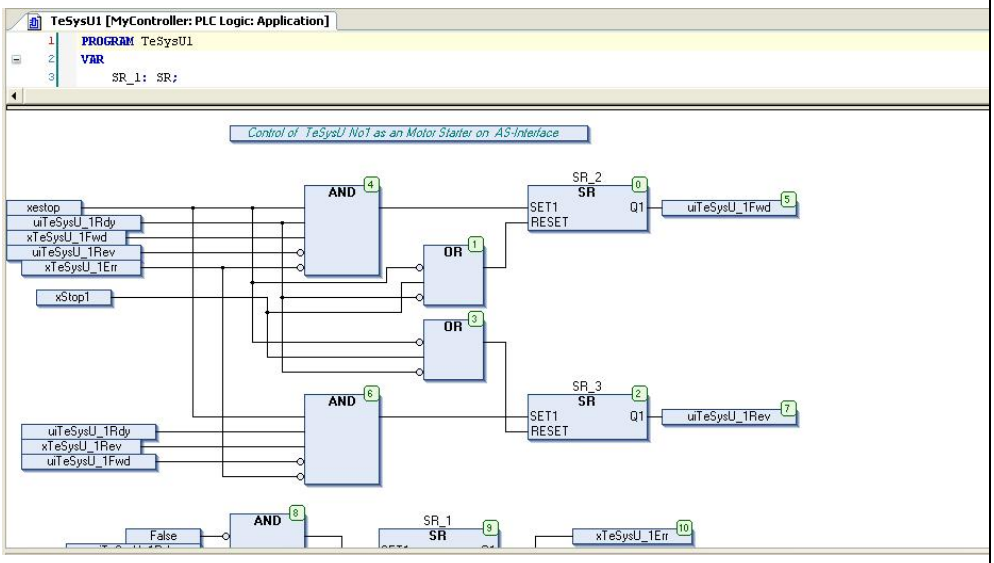
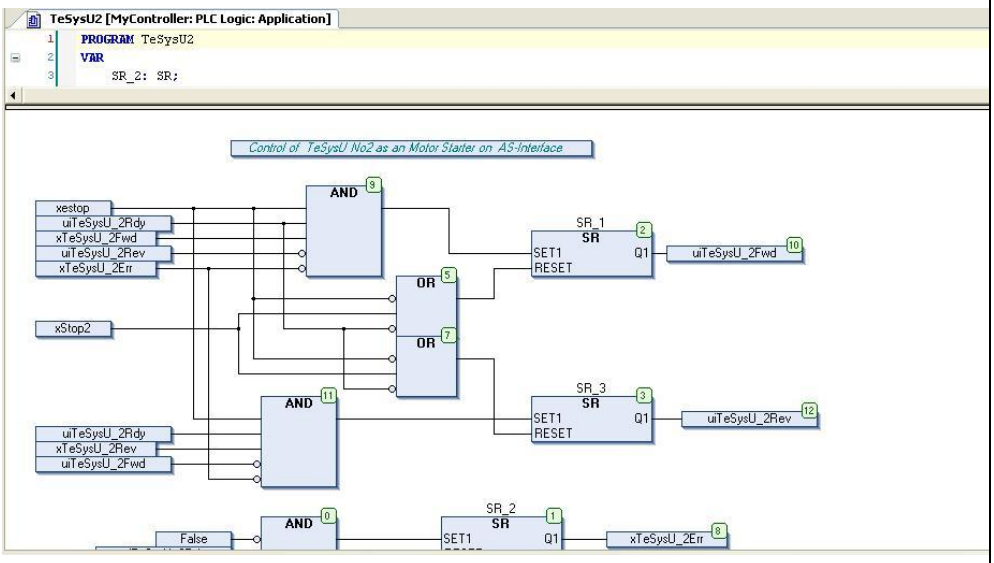
Login to controller

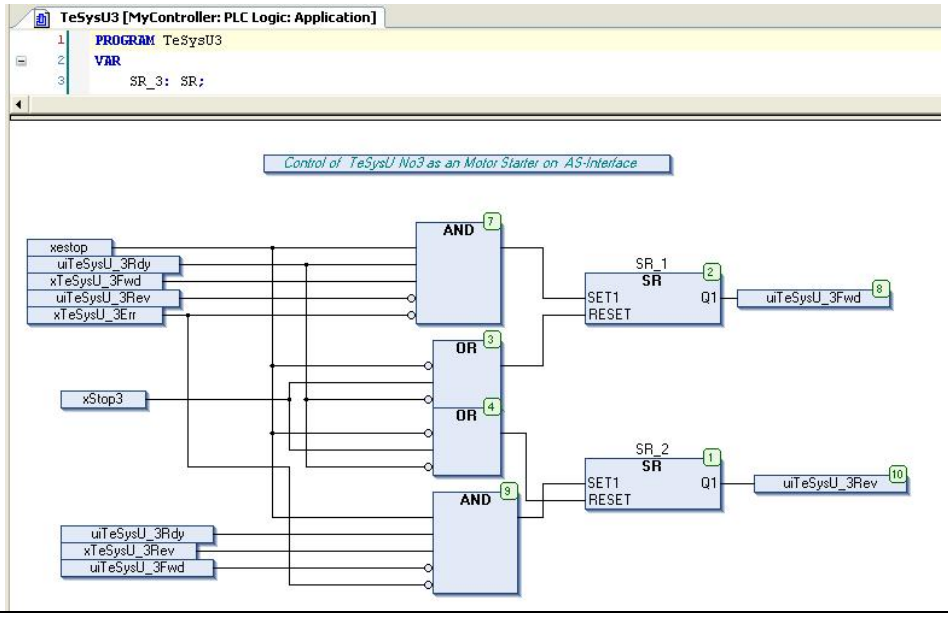
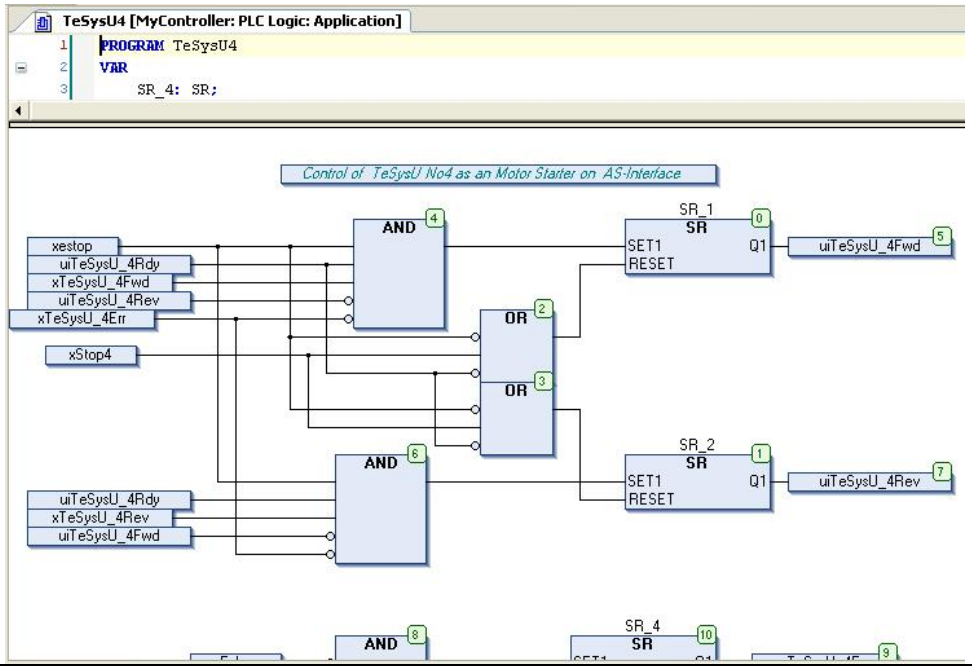
1	To login to the controller click **Online→ Login**	
2	If the controller program is different from the program on the PC a message asks you if you wish to replace the old controller program. Select the operation you want and press **OK** to confirm the download**.**	
3	The actual download status is displayed at the bottom left corner of the main window.	
4	To start the new Application select **Online→ Start**	

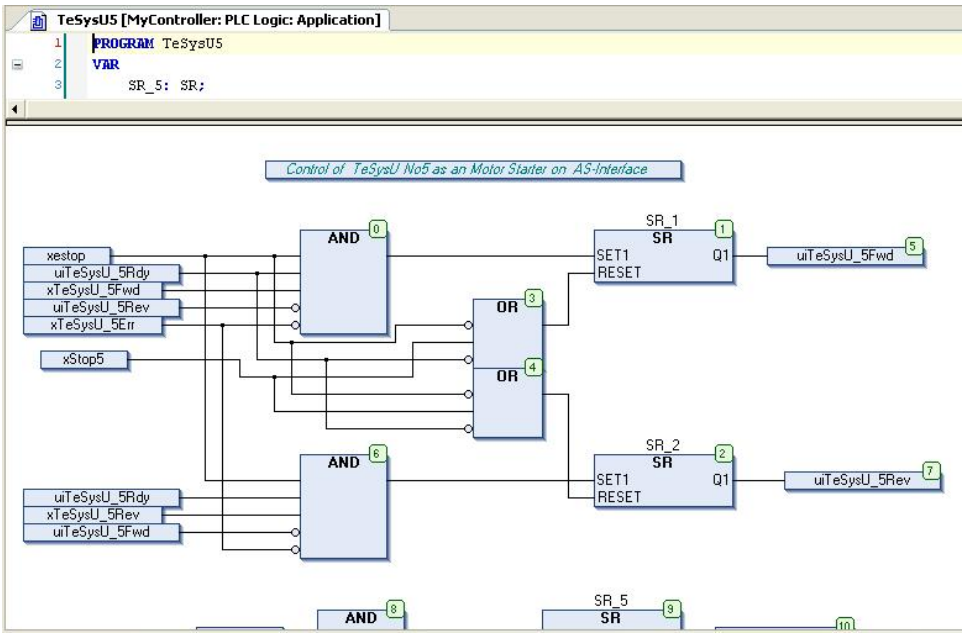
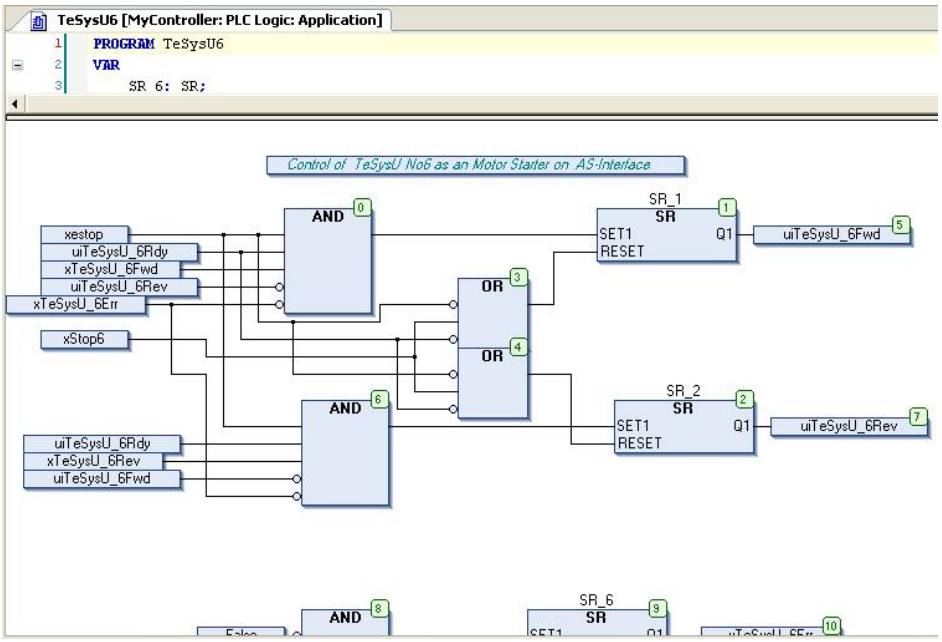
5	If everything is operating normally the devices and folders are marked in green otherwise they will be marked in red.	

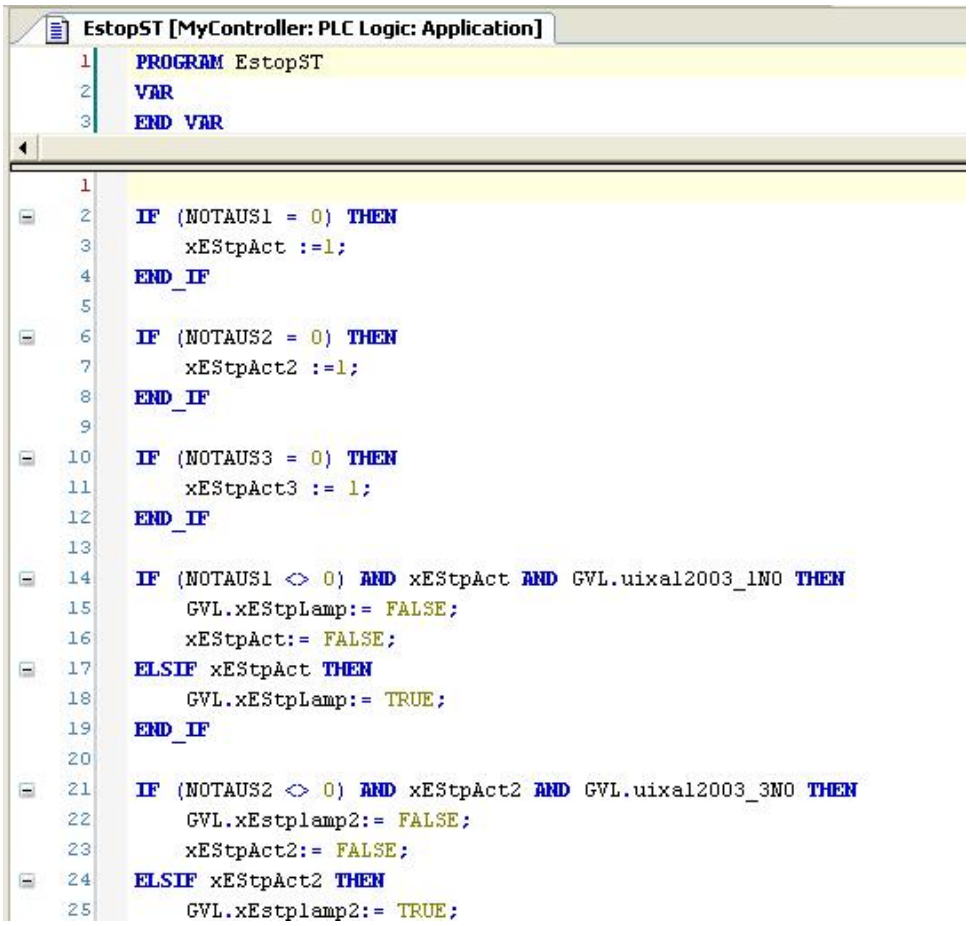
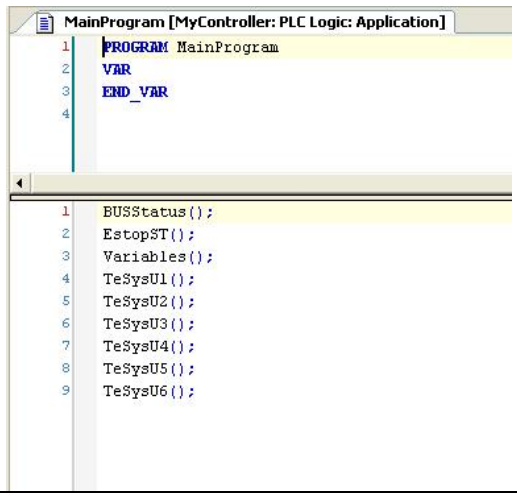
Application Overview

1	The right picture shows the structure of the program. Every function has an own point in the structure.	
2	**POU BUSStatus** shows the FBs to read the status of Master and Slave on ASi Network 	

3	**POU TeSysU1** shows the control of TeSysU1 
4	**POU TeSysU2** shows the control of TeSysU2 

5	**POU TeSysU3** shows the control of TeSysU3 
6	**POU TeSysU4** shows the control of TeSysU4 

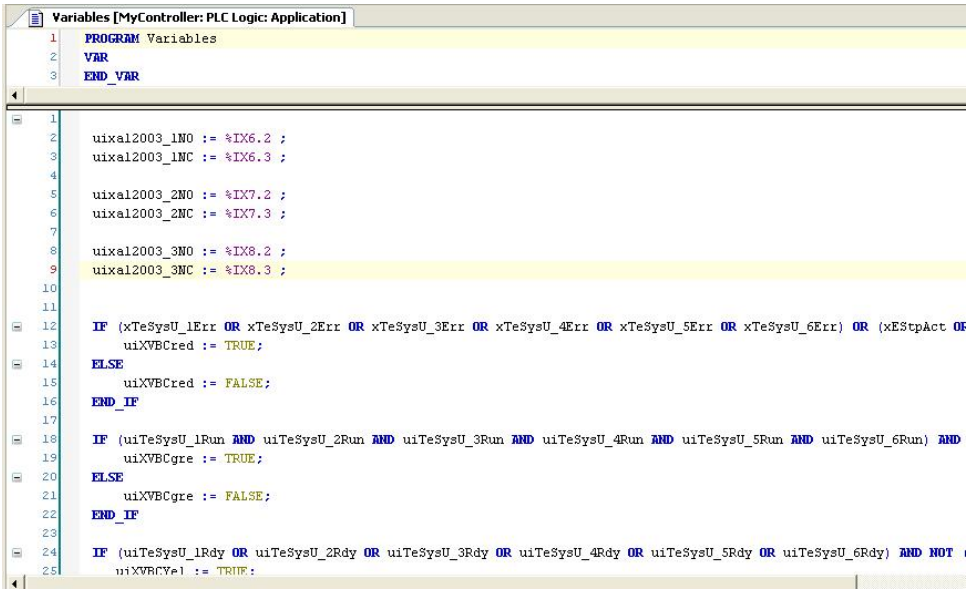
7	**POU TeSysU5** shows the control of TeSysU5 
8	**POU TeSysU6** shows the control of TeSysU6 

9	**POU EstopST** shows the logic of Emergency Stop 
10	**POU MainProgram** shows the main program 

```
EstopST [MyController: PLC Logic: Application]
PROGRAM EstopST
VAR
END_VAR

IF (NOTAUS1 = 0) THEN
    xEStpAct :=1;
END_IF

IF (NOTAUS2 = 0) THEN
    xEStpAct2 :=1;
END_IF

IF (NOTAUS3 = 0) THEN
    xEStpAct3 := 1;
END_IF

IF (NOTAUS1 <> 0) AND xEStpAct AND GVL.uixal2003_1NO THEN
    GVL.xEStpLamp:= FALSE;
    xEStpAct:= FALSE;
ELSIF xEStpAct THEN
    GVL.xEStpLamp:= TRUE;
END_IF

IF (NOTAUS2 <> 0) AND xEStpAct2 AND GVL.uixal2003_3NO THEN
    GVL.xEstplamp2:= FALSE;
    xEStpAct2:= FALSE;
ELSIF xEStpAct2 THEN
    GVL.xEstplamp2:= TRUE;
```

```
MainProgram [MyController: PLC Logic: Application]
PROGRAM MainProgram
VAR
END_VAR

BUSStatus();
EstopST();
Variables();
TeSysU1();
TeSysU2();
TeSysU3();
TeSysU4();
TeSysU5();
TeSysU6();
```

11	**POU Variables** shows Alarm handling



```
Variables [MyController: PLC Logic: Application]
PROGRAM Variables
VAR
END_VAR

uixal2003_1NO := %IX6.2 ;
uixal2003_1NC := %IX6.3 ;

uixal2003_2NO := %IX7.2 ;
uixal2003_2NC := %IX7.3 ;

uixal2003_3NO := %IX8.2 ;
uixal2003_3NC := %IX8.3 ;


IF (xTeSysU_1Err OR xTeSysU_2Err OR xTeSysU_3Err OR xTeSysU_4Err OR xTeSysU_5Err OR xTeSysU_6Err) OR (xEStpAct OR
    uiXVBCred := TRUE;
ELSE
    uiXVBCred := FALSE;
END_IF

IF (uiTeSysU_1Run AND uiTeSysU_2Run AND uiTeSysU_3Run AND uiTeSysU_4Run AND uiTeSysU_5Run AND uiTeSysU_6Run) AND
    uiXVBCgre := TRUE;
ELSE
    uiXVBCgre := FALSE;
END_IF

IF (uiTeSysU_1Rdy OR uiTeSysU_2Rdy OR uiTeSysU_3Rdy OR uiTeSysU_4Rdy OR uiTeSysU_5Rdy OR uiTeSysU_6Rdy) AND NOT
   uiXVBCYel := TRUE;
```

HMI

Introduction

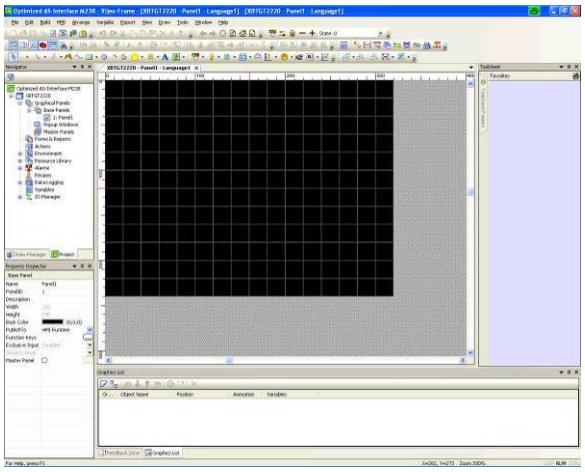
This application uses a HMI device of type Magelis XBGT2220. The HMI communicates with the controller using SoMachine protocol over serial port (RS485). The Magelis is programmed using the **Vijeo Designer** software tool (delivered with SoMachine), that is described in briefly the following pages. For the connection between the controller and the HMI the cable **XBTZ9008** is used.

NOTE:
The Vijeo Designer Tool is opened through SoMachine. For more information see Chapter controller->Add Vijeo Designer HMI

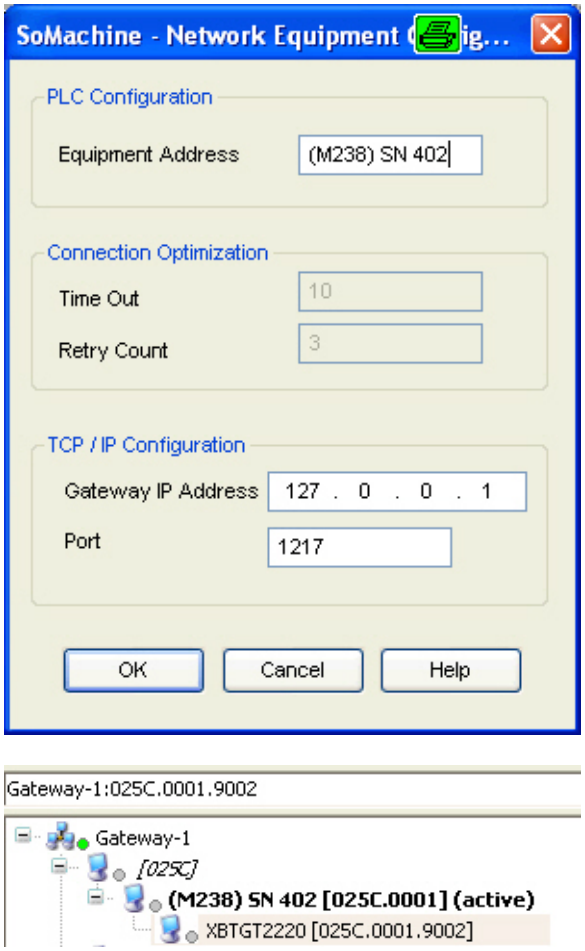
Setting up the HMI is done as follows :

- Main window
- Communication settings
- Create a Switch
- Create a Lamp
- Create a Numeric Display
- Example screens

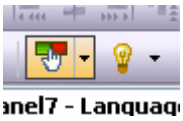
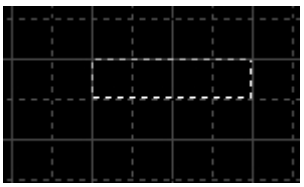
Main Window

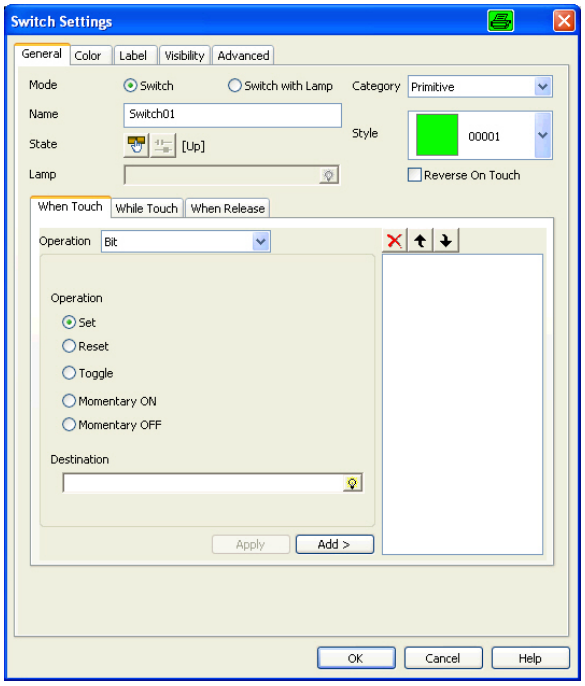
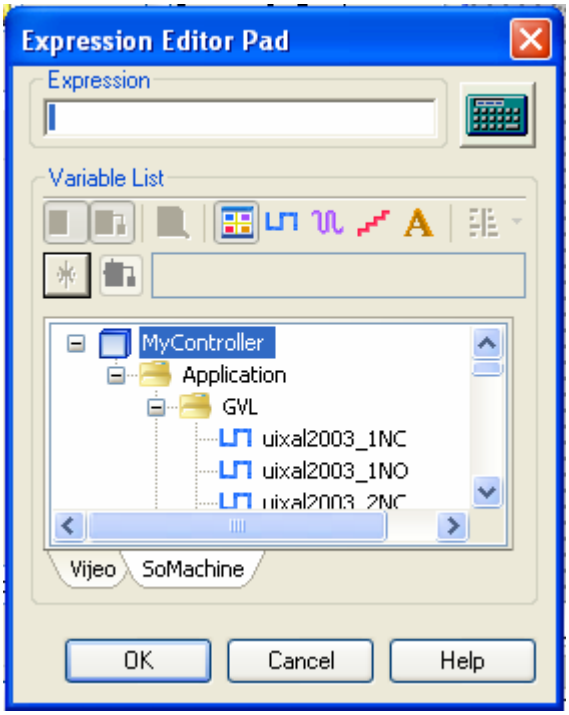
1	After creating a Vijeo Designer HMI in SoMachine the main Window of Vijeo Designer opens.	
2	With these new variables Vijeo Designer creates a **SoMachineNetwork01** for the communication with the controller. Double click on: **SOM_XBTGT2220**	

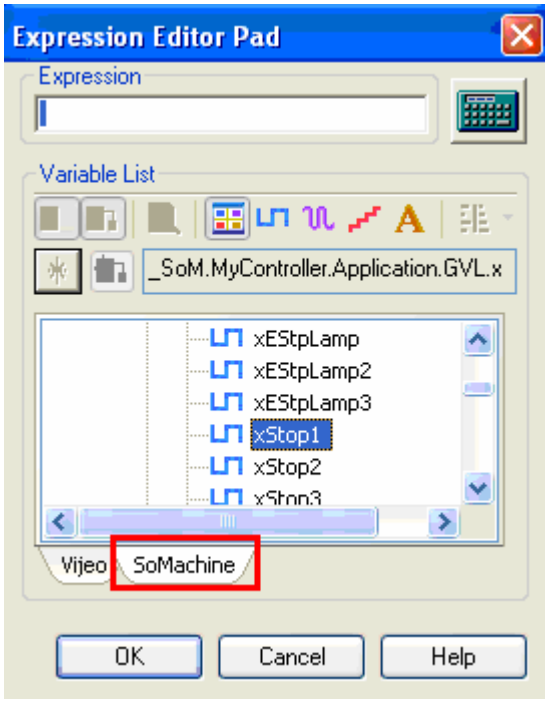
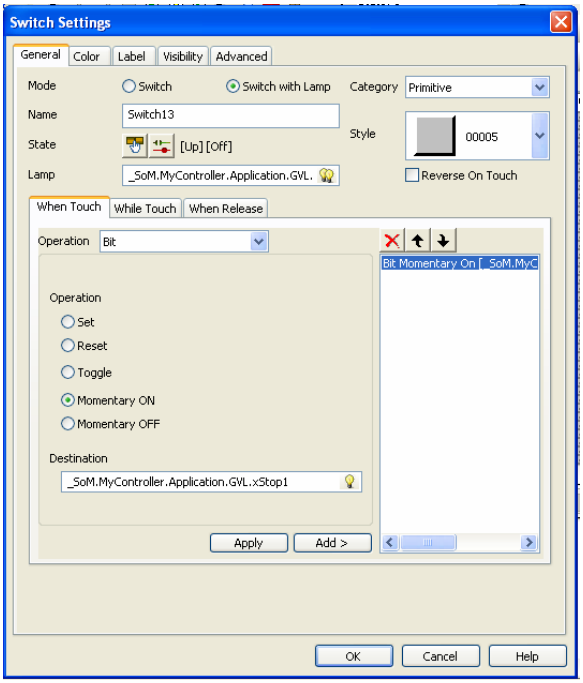
Communi-cation settings

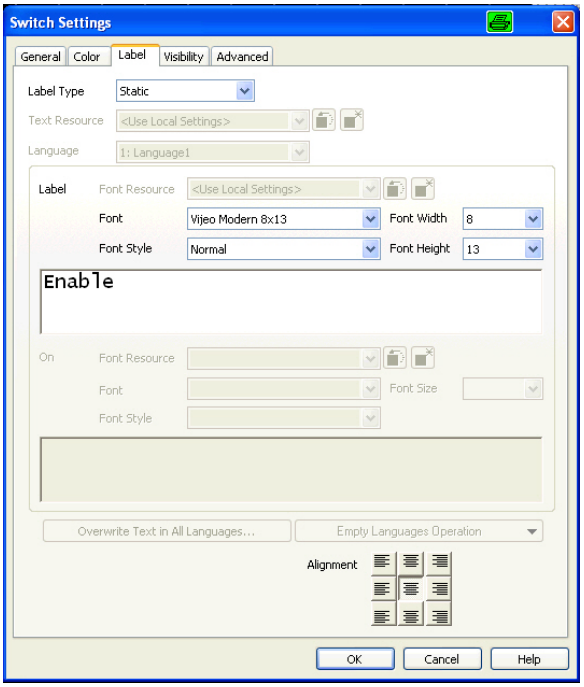
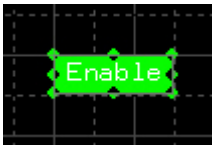
3	…and enter the controller name under **Equipment Address**, here in the example **(M238) SN 402.** **Note:** **The serial number of the M238 controller is on the label of the front side flap. It is a unique number.** The name of the controller is displayed in the **Communication settings** folder in SoMachine. In our Project it is (**M238) SN 402.** Click on **OK**.	

Create a switch

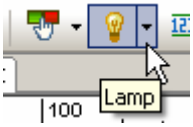
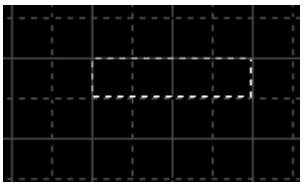
1	To connect a controller variable to a switch object: Select the **Switch** button in the **Menu bar**	
2	Select the position of the switch on the screen by opening the rectangle. Fix the size of the switch by altering the size of the rectangle and press **enter.**	

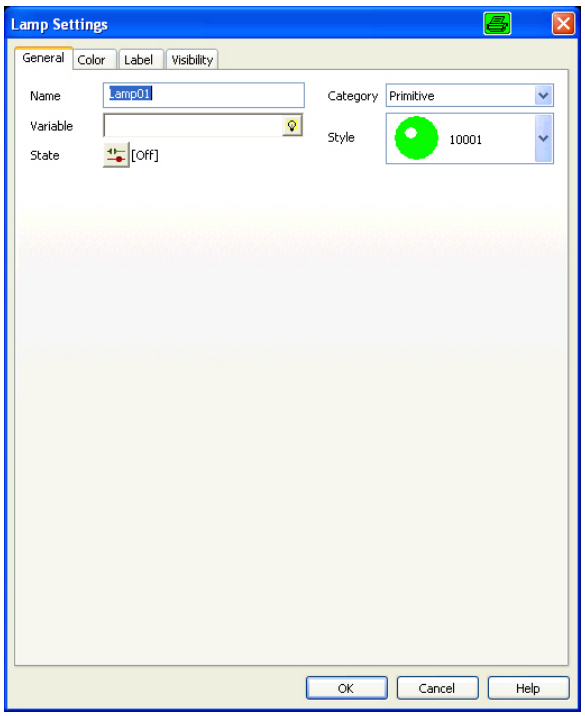
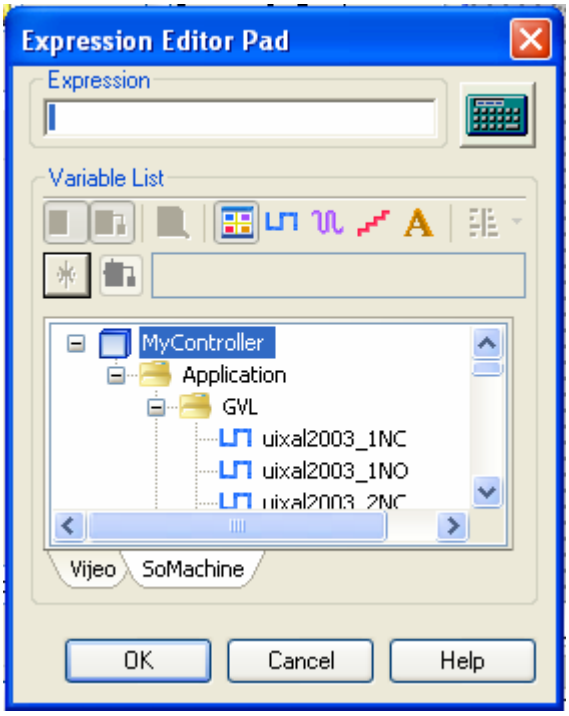
3	In the **Switch Settings** dialog, select the variable that should be linked to the button (you can browse for a variable by clicking on the bulb icon at the end of the edit box).	
4	Select the **SoMachine** tab , **→MyController** **→ Application** **→GVL.**	

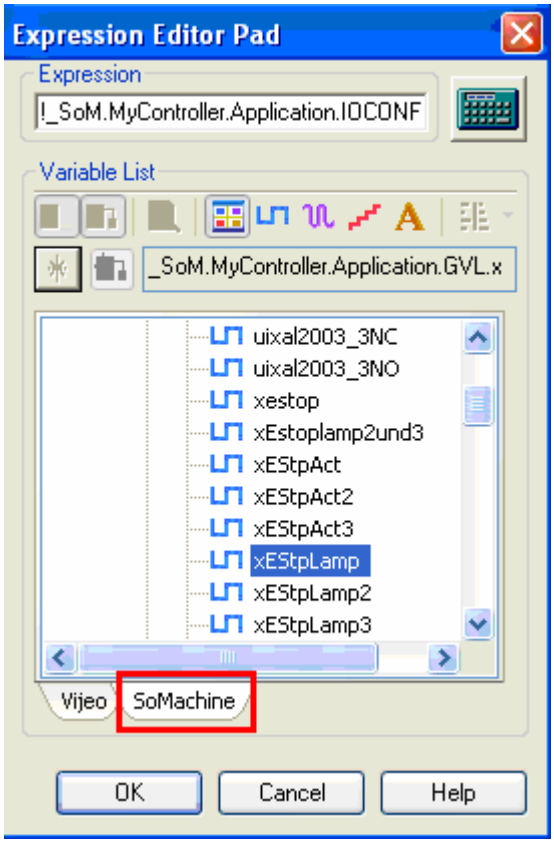
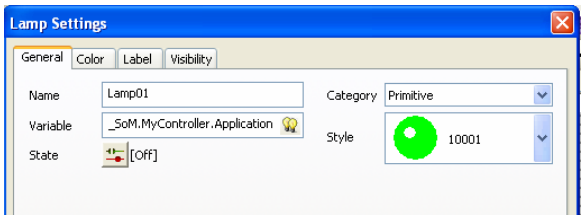
5	Open the **GVL** directory and select the required boolean variable (e.g. xStop1). Click on **OK.**	
6	The new switch variable is set in the **Destination** field. To finish the action click **OK**	

7	Go to the **Label** tab. Here select **Label Type: Static** and enter a name for the button, e.g. enable. Once you have finished your settings click on **OK**.	
8	The new switch is now displayed on the work top.	

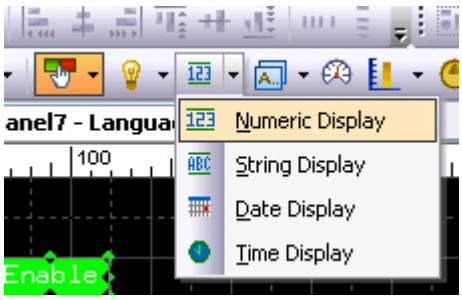
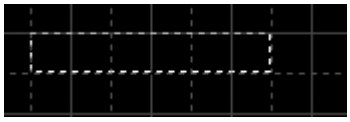
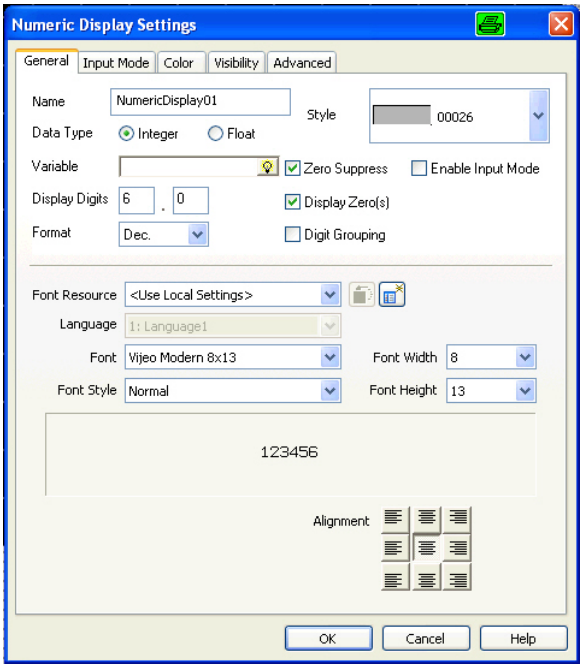
Create a Lamp

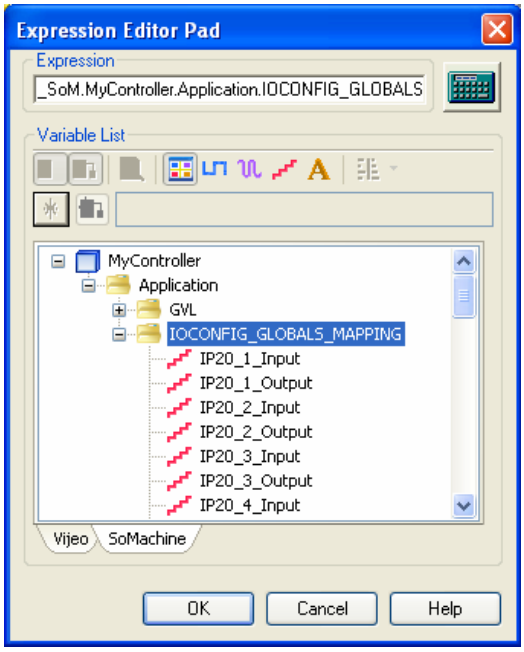
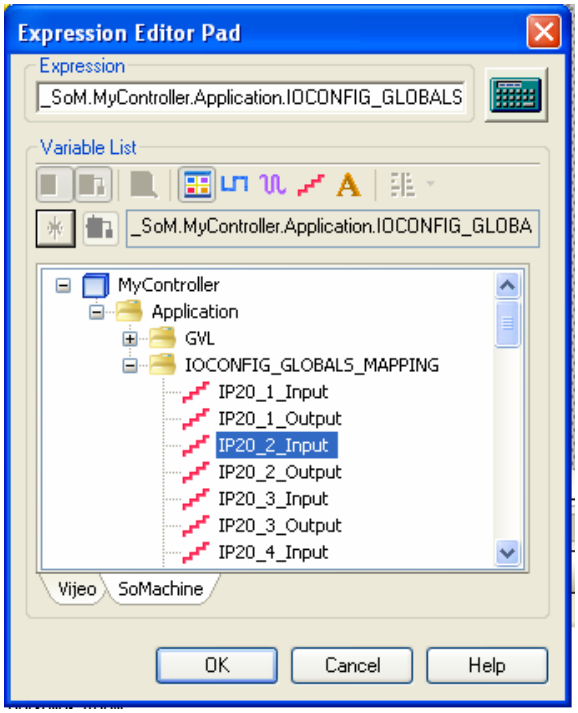
1	Select the **Lamp** button in the **Menu bar**	
2	Select the place where you want to place the button by opening the rectangle and pressing **enter.**	

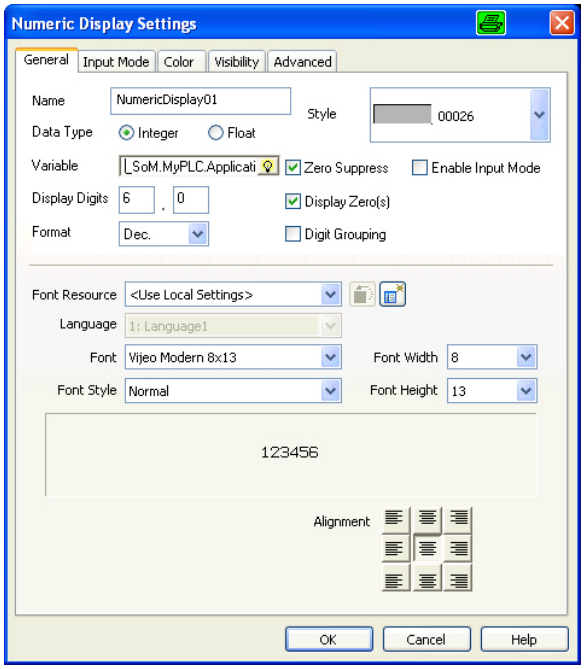
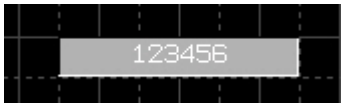
3	In the **Lamp Settings,** select the variable that should be linked to the button (**bulb icon**).	
4	Select the **SoMachine** tab , →**MyController** → **Application** →**GVL..**	

5	Then open the **GVL** directory and select the needed boolean variable (e.g. xEStpLamp) and click **OK**	
6	The new lamp variable is set in the **Variable** field. To closed the action click **OK**	
7	On the Work frame is now the new lamp.	

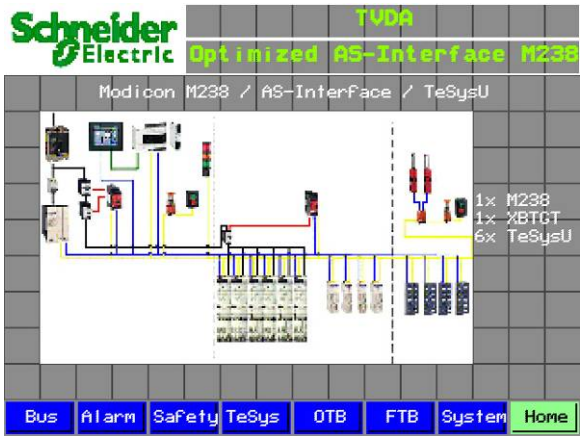
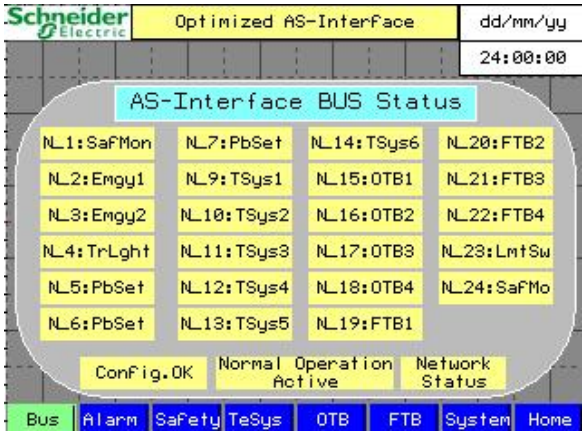
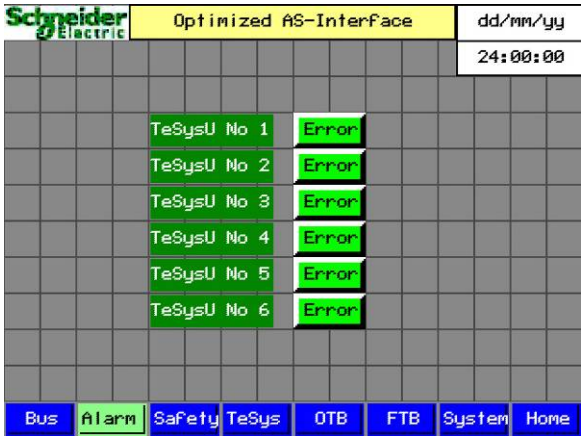
Create Numeric Display

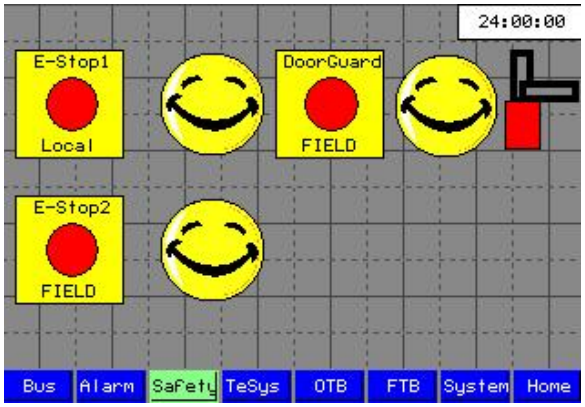
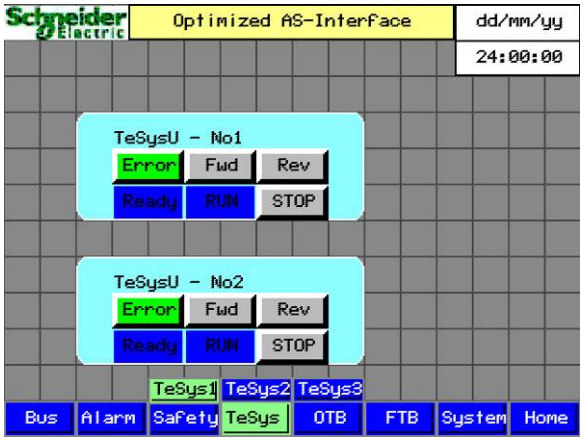
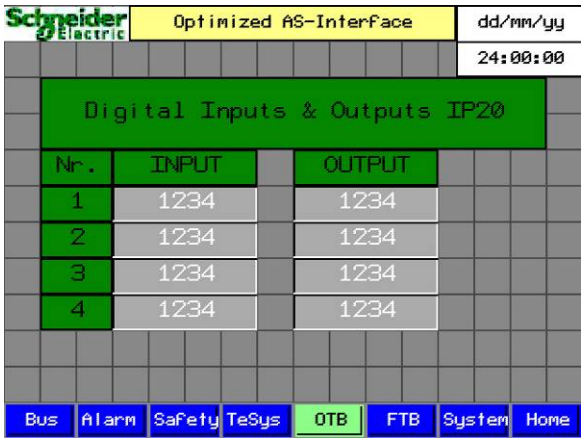
1	Click on the **Numeric Display** icon in the tool bar.	
2	Select the position where you want to place the display by opening the rectangle and pressing **Enter.**	
3	In the **Numeric Display Settings** dialog go to the **General** tab. In **Display Digits** you can set the maximum number of the digits to be displayed for integral and fractional part of the value. To link a **Variable** to the display click on the **bulb icon** to browse for a variable**.**	

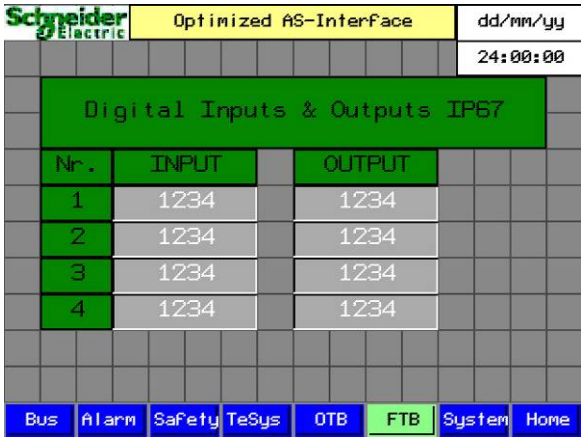
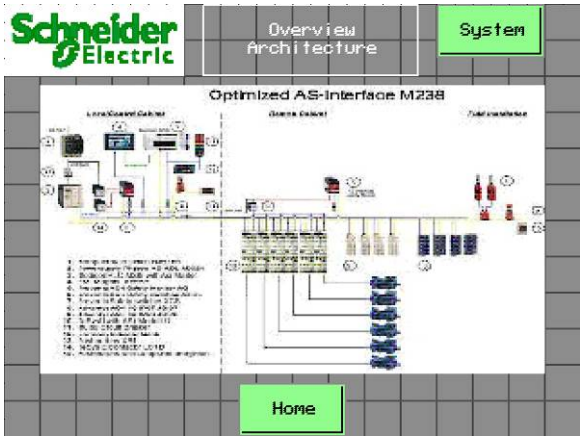
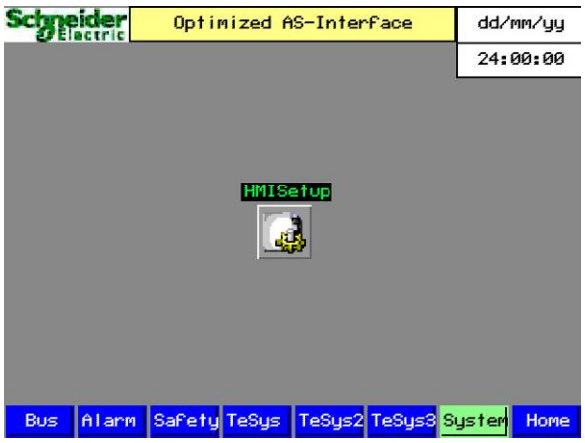
4	Select the **SoMachine** tab , →**MyController** → **Application** → **IOCONFIG_GLOBALS_MAPPING** →	
5	Then open the **IOCONFIG_GLOBALS_MAPPING** directory and select the needed bool variable (e.g. State) and click **OK**	

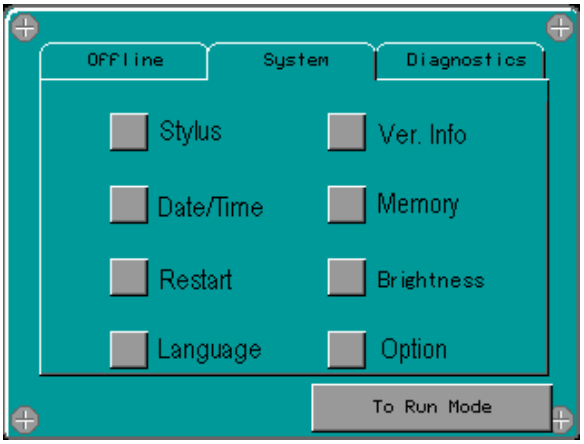
6	The new Numeric Display variable is set in the **Variable** field. To closed the action click **OK**	
7	On the Work frame is now the new numeric display shown.	

Example screens

1	The **Home** page shows a picture of the complete architecture.	
2	The **Bus** page shows the AS-Interface state of every device. Green for ready. Red for fault.	
3	Page **Alarm** shows the current alarm state for the TeSysU	

4	The "**Safety**" page shows the state of the Emergency Stop relay.	
5	The **TeSys1**, **TeSys2** and **TeSys3** pages are used for parameterization and control of TeSysU motor starters. Each page contains two starters.	
6	The **OTB** page shows the I/O states of the 4 OTB's installed in the main cabinet. The Inputs and the Outputs are displayed as a Byte.	

7	The **FTB** page shows the I/O states of the 4 FTB's installed in the field. The Inputs and the Outputs are displayed as a Byte.	
8	The **Overview Architecture** page includes the link to the **System** page and **Home.** This page will be shown if the user click in the **Home** page on the architecture picture.	
9	The **System** page is for setting the HMI parameters.	

10	Return with **To Run Mode**.	

AS-Interface device addressing

Introduction

This chapter describes the steps required to initialize and configure the different devices required to attain the described system function.

General

The AS-Interface bus configuration within SoMachine is used to configure:

- AS-Interface Master for the M238
- AS-Interface Slaves

This is described in chapter Controller.

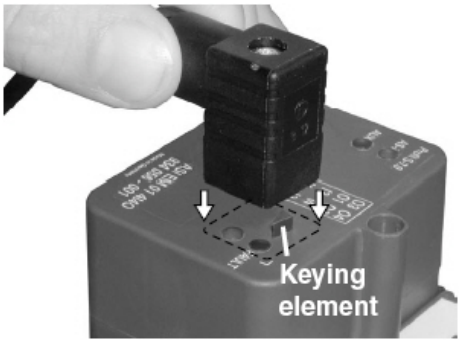
Additional the AS-Interface addressing of each AS-Interface slave is done by using the ASITERV2 handheld and the dedicated adapter cable.

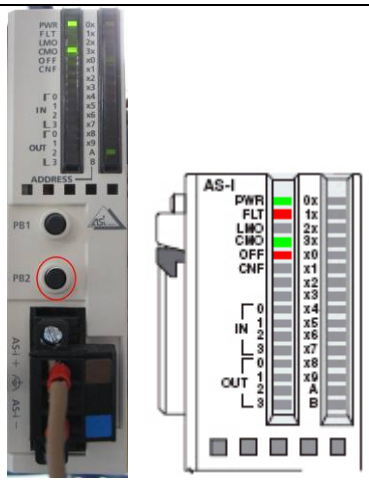
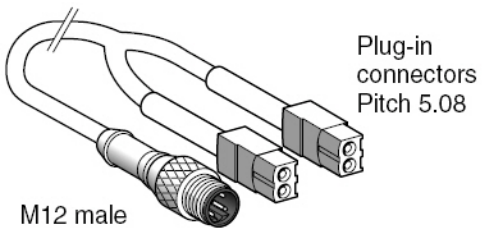
Equipment for Node Addressing

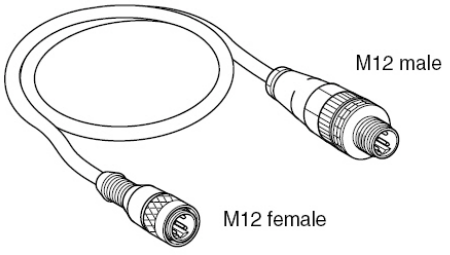
***AS*/TERV2 Handheld**

1	Use the ASITERV2 handheld to address the individual slaves.	

AISSL*
Safe Input Slaves

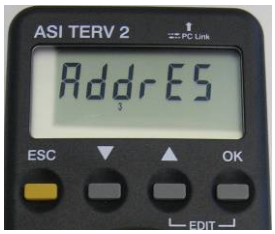
2	E.g. addressing of an E-Stop **ASISSLB4 + ZB4BS844**	
	Use the **ASITERIR1** infrared adapter cable to address safe input slaves. Please note the coding key on the slave and connect the adapter to the node.	 

M238 AS-Interface Master module TWDNOI10M	3	**The M238 AS-Interface Master module use Auto addressing mode (in SoMachine parameterized)** In order for slave addressing to be successful, the nodes must be connected to the power supply via the yellow AS-Interface cable. Before you start addressing, switch the master 'offline' by pressing and holding down the PB2 button on the AS-Interface master module for 3 – 4 seconds. The master will switch to offline mode and will indicate this on the module via an LED lighting up red next to the word OFF.	
ASILUFC5 ASI20M*	4	Use the **XZMG12** adapter cable to address ASI20M* and **ASILUFC5** IP20 devices (TeSysU motor starters communication cassette)). Connect the adapter to the node via the yellow plug. On slaves requiring a 24 V auxiliary supply (black AS-Interface cable) at output level (e.g., as is the case with TeSysU), the power supply must be connected when the nodes are programmed.	  **M12 male with yellow and green plug-in connectors**

ASI 20MT4I3OSE	5	Use the **ASITERACC2** adapter cable to address IP20 field devices. Connect the adapter to the M12 female AS-Interface AUX on the right underside of the device.	
ASI67FMP43E	6	Use the **ASITERACC1F** adapter cable to address IP67 field devices. Connect the adapter to the M12 female AS-Interface AUX on the right underside of the device.	 

Addressing of AS-Interface Slaves

1	To perform addressing, turn the rotary switch to the ADDR position and press the OK button on the top right. The device will now look for connected nodes and display the address of any slaves it locates within a few seconds.	
2	If no AS-Interface appears on the display, the device has not been able to locate any AS-Interface nodes and you should check the connection between the addressing device and the slave.	
3	The address of the connected slave will appear on the display as read by the device. The factory setting for new slaves is 0.	
4	You can press the two arrows in the center of the device (up/down) to set the address in the range from 0 to 31 (0 is not a valid slave address). While an address is being set, the current address will continue to appear on the display. On slaves with advanced addressing, an A or B will appear on the display after the address, indicating the channel assignment. In this example, both the IP20 (ASI20M*) and IP67 (ASI67FMP*) modules support advanced addressing. *The photo shows an ASI20MT4I3OSE module with address 12A.*	
5	Once you have set the required address, press OK to apply the setting. During transmission, the display will switch to ProG.	

6	Once you have made the address setting, the new address will appear permanently on the device display. To address another device, press the ESC button on the left-hand side and resume the process at Step 4.	

Safety Monitor

Introduction

This section describes how to parameterize, load, start, and stop the safety monitor by using the ASIWIN software.

In order to transfer and also enable, start and stop an application on the safety monitor, a unique password must be entered, thereby helping to prevent against unauthorized access.

As an additional measure, the RJ45 programming port on the safety monitor can also be capped using the transparent plastic stoppers supplied with the product and sealed to prevent access. The breaking of a seal or removal of a stopper would indicate tampering.

The ASIWIN software, which is described below, is used to create the application on the safety monitor.

The Preventa AS-Interface safety monitor supervises the functional safety of the system. The safety monitor replaces functions previously normally implemented with Preventa modules of the XPSAC series (or similar), and conforms to functional safety standards.

Advantage of using an AS-Interface solution:

- Inputs/Outputs from both standard and safety slaves can be transmitted over the same bus cable
- Even the most remote safety device (e.g. E-Stop) can be simply snapped onto the yellow bus cable, instead of having to lay down extra cabling.

Due to individual configuration, applications with 2 or more safety monitors allow multiple use of safety devices(e.g. E-Stop) by overlapping the safety zones.

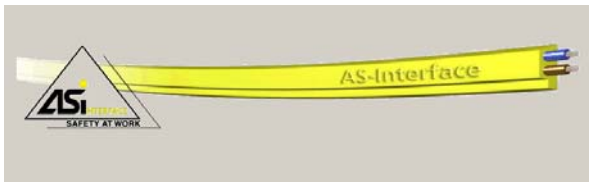
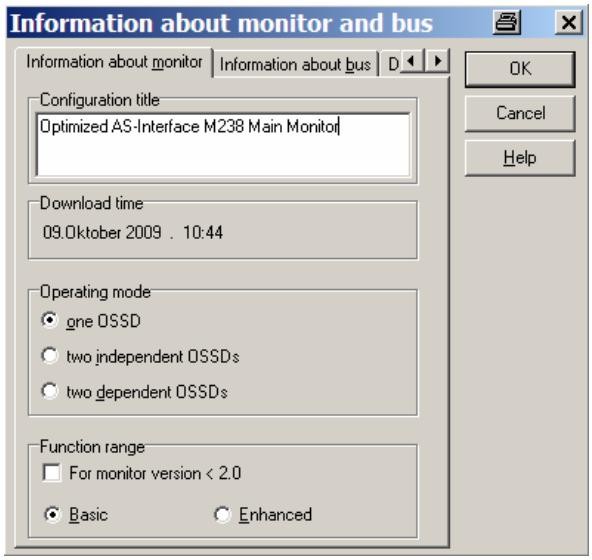
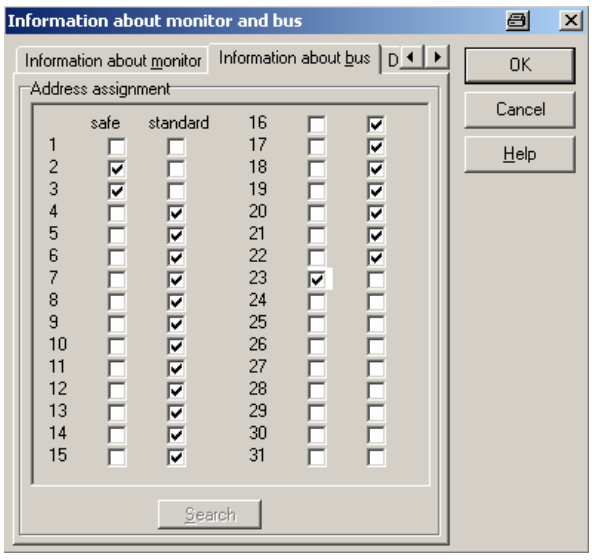
Preconditions

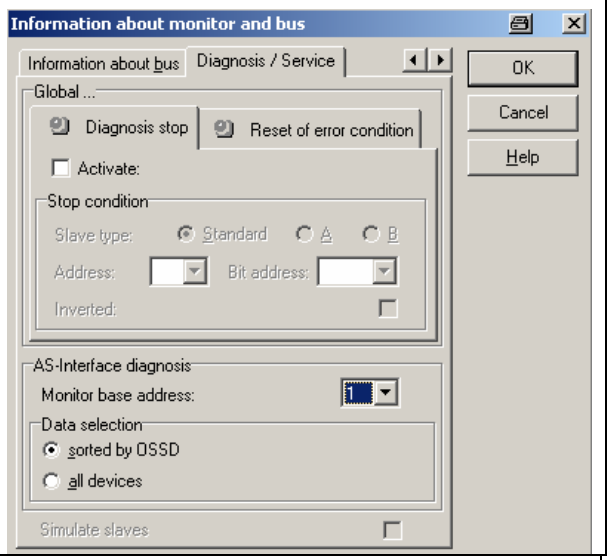
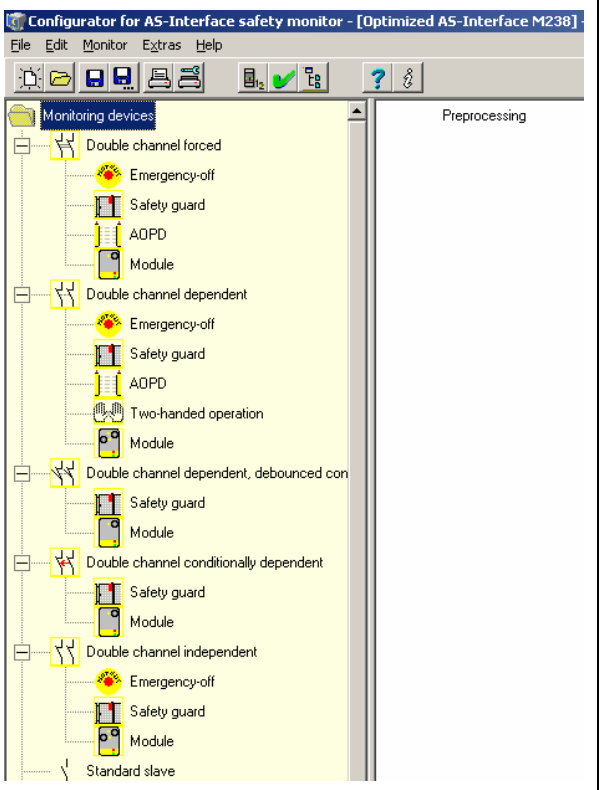
The procedure described below is subject to the following prerequisites being met:

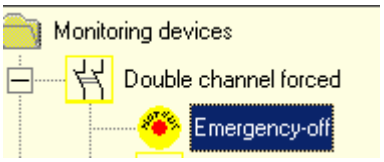
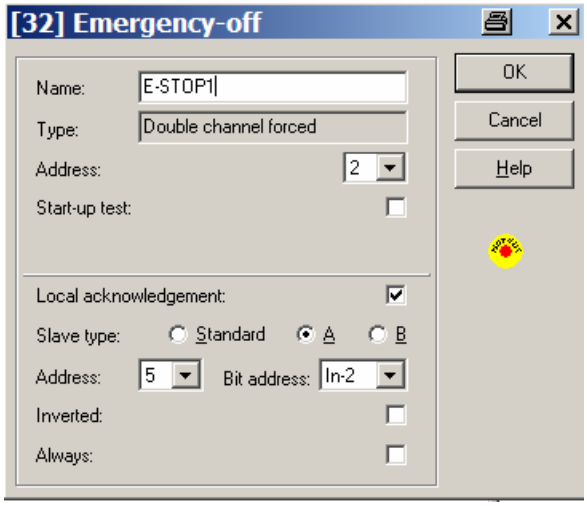
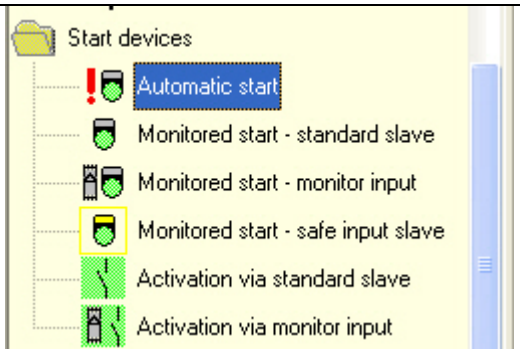
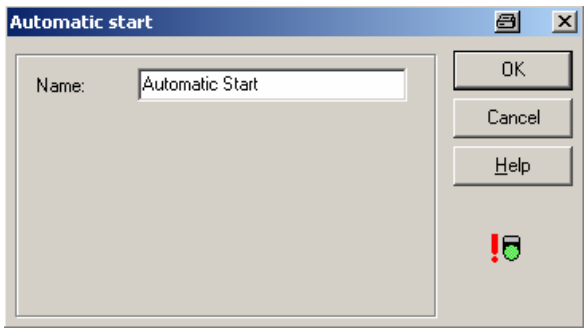
- The ASIWIN software installed on your PC.
- The safety monitor is connected to the power supply and to the AS-Interface master on the M238 (via the AS-Interface cable).
- The functional safety AS-Interface slave devices (see Component List) and the standard AS-Interface slave devices slaves used for functional safety (e.g., for acknowledgement, error reset) are correctly addressed and ready for operation on the AS-Interface network.
- The PC is connected to the Preventa Safety Monitor (ASISAFEMON1) via the special interface cable (ASISCPC).
- The password for configuring the safety monitor is known (the factory default password is “SIMON”).

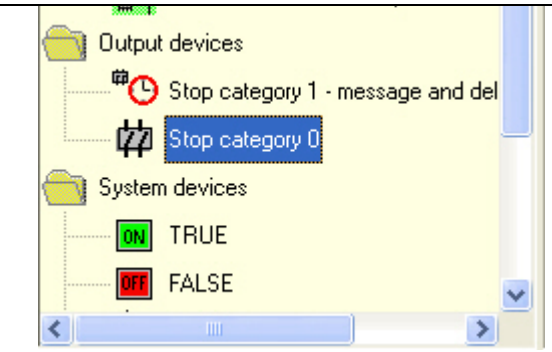
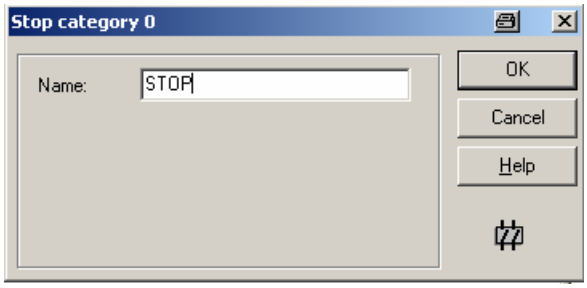
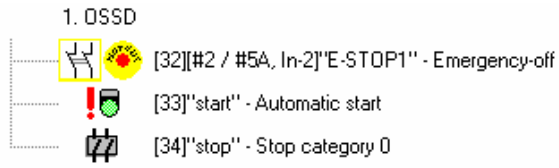
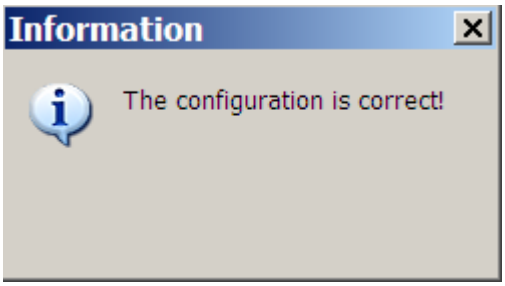
Remark

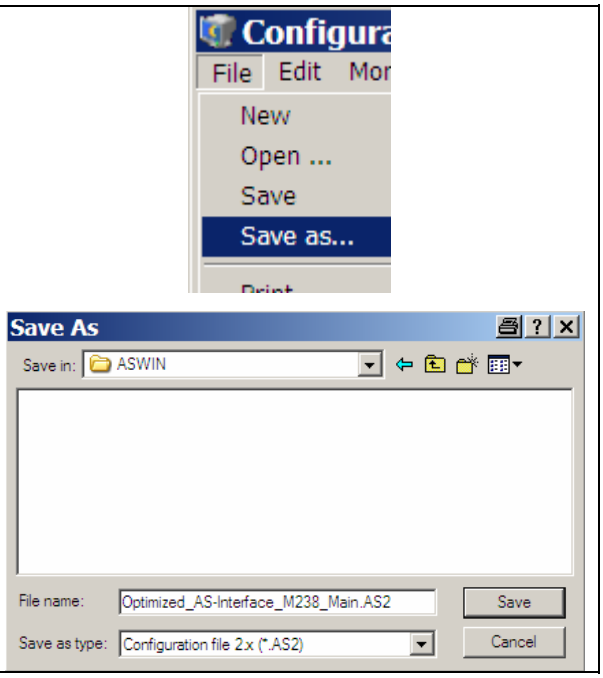
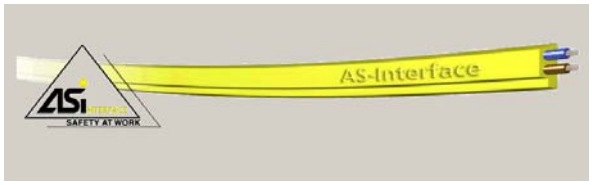
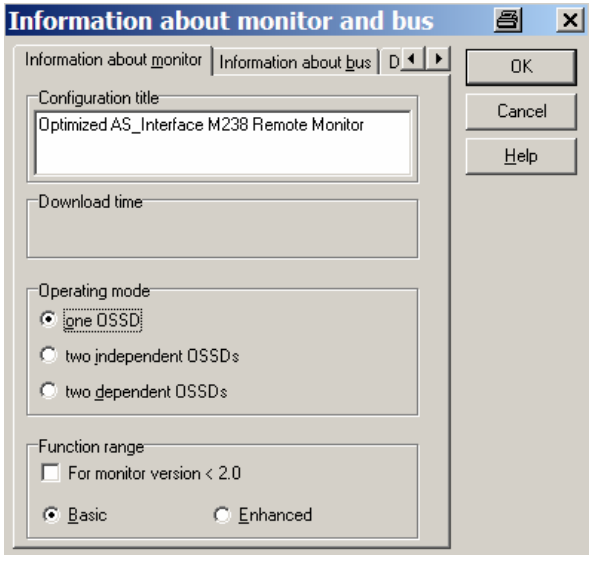
In our example we have two different safety zones, one zone for the Main cabinet and one zone for the Remote cabinet. The zones are monitored by two different Safety monitors. For the two zones you need to create separate configurations for both Safety Monitors.

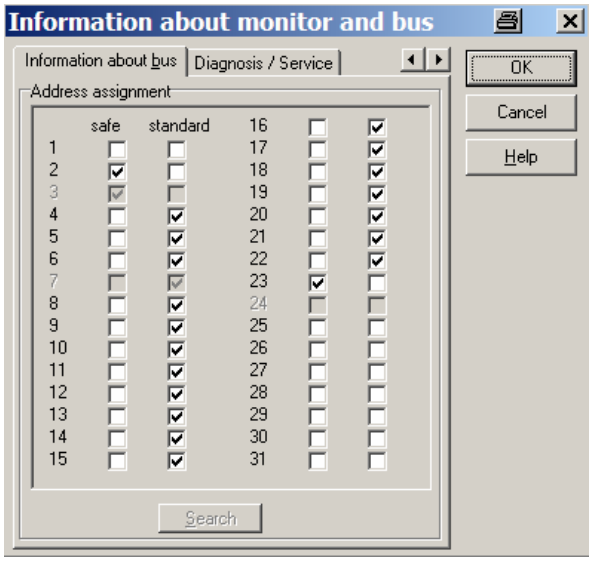
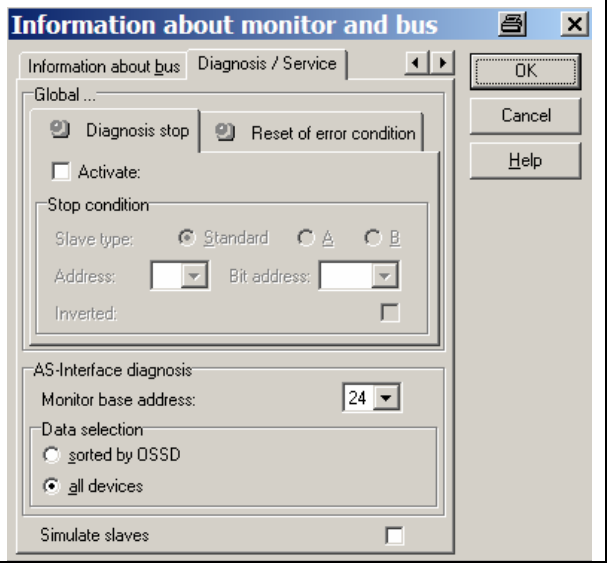
Starting ASIWIN	1	When the ASIWIN software starts up, the screen opposite will appear, offering you the following options (some of which will be grayed out): • Diagnostics • New configuration • Open configuration • Load configuration from AS-Interface safety monitor If this dialog box does not appear, use **New** or **Open** in the **File menu.**	
Create a New Configuration Setting up the Monitor for the MAIN cabinet	2	To create a new configuration, first assign a unique title to your configuration on the **Information about monitor** tab. In this example, an **ASISAFEMON1**, which has just one **OSSD** and basic functionality, is being used.	
Select the tab Address assignment Fill out the node types safe and standard	3	On the **Information about bus** tab, enter the safe and standard slaves you are using. If you are using standard slaves for acknowledgement and error reset, you will need to enter them here. The slave addresses of Preventa inputs are 2, 3 and 23. The safety monitors for which addresses 1 and 24 have been reserved are not entered here.	

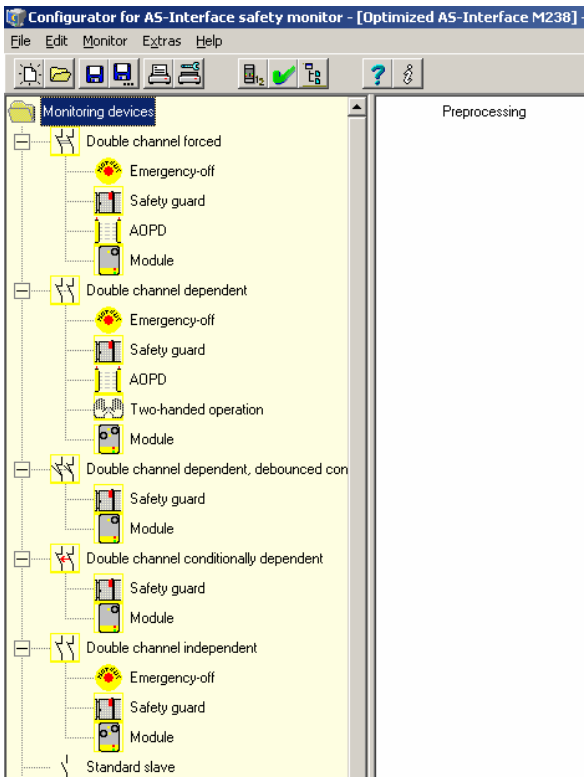
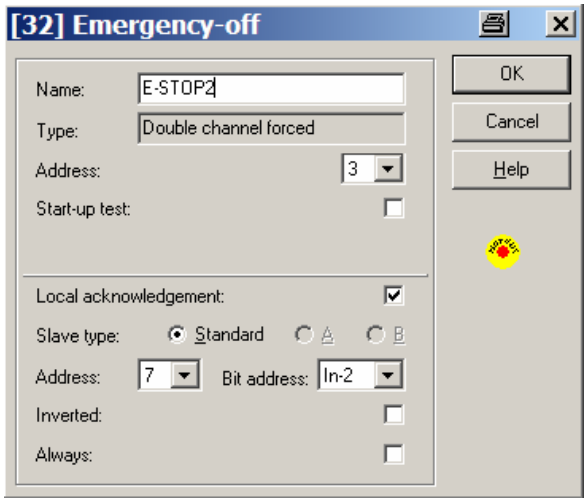
Select the Diagnosis / Service tab Fill out the Monitor base address	4	Enter the address of the safety monitor on the last tab, **Diagnosis / Service.** The address 1 is used for the safety monitor of the Main cabinet. Confirm with **OK.**	
Available Monitor functions	5	The screen on the right will appear for the base controller. A variety of functions can be implemented, these include: • Emergency-off • Safety guard • Module • Feedback loop • Monitored start To use the individual blocks, drag the elements from the yellow area on the left-hand side to the white area on the right-hand side **(1. OSSD).**	

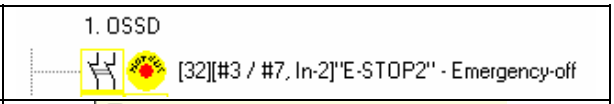
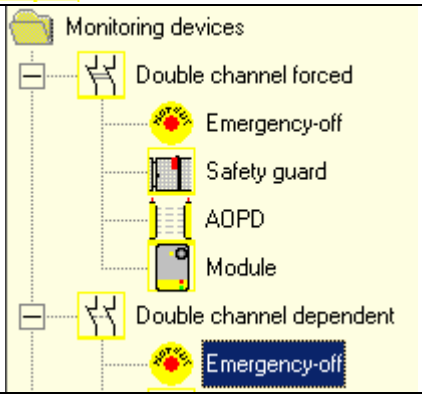
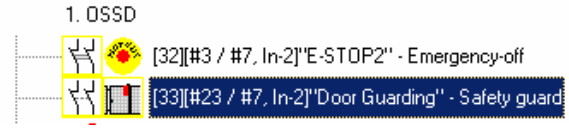
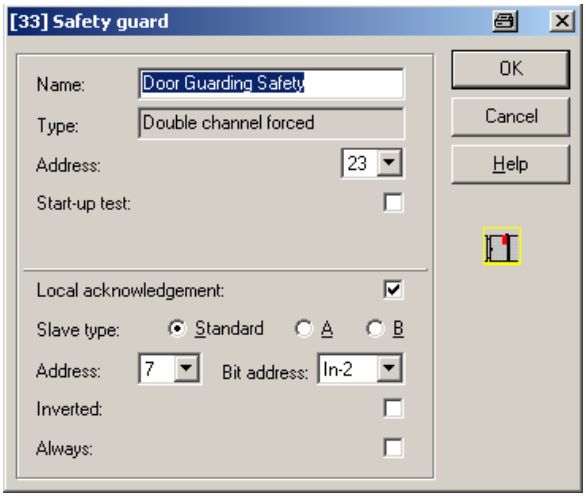
Adding an Emergency-Stop	6	The following dialog box will appear if you place a forced Emergency-off in the area. In the **Name** field, enter a unique name, which will help you to find the device reliably and without confusion (all device names should be selected on this basis). Next, assign an **Address** to the device; only the addresses you entered as safe slaves when configuring the motor and have not yet used will appear. We use Local acknowledgement. In our example we use the AS-Interface Slave address 5 which depends to one of the XALD Pushbuttons. The Bit address is IN-2. Note: If you activate the Start-up test, when the power supply is restored, the device specified must be forced (triggered) in order to be able to acknowledge the monitor. Confirm with **OK.**	 
Selecting the Start Condition	7	Once the switches and safety guard monitoring have been added to the configuration, the start device is added. In the example we use Automatic Start. Confirm with OK.	 

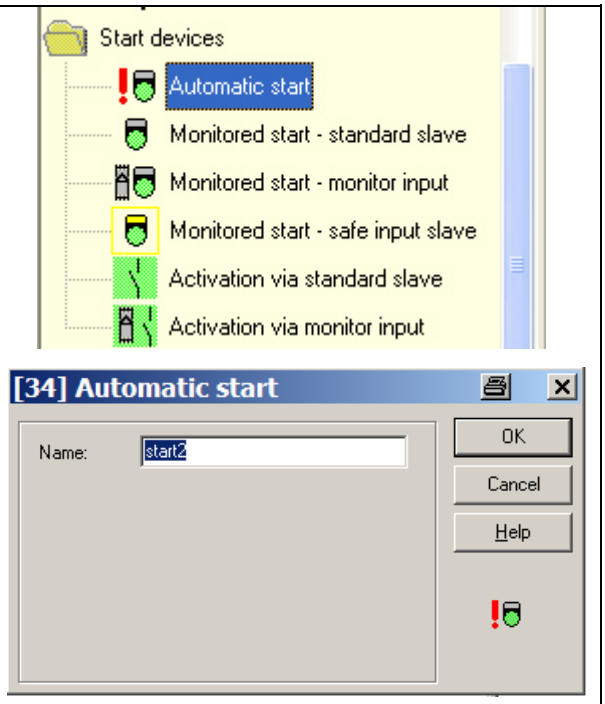
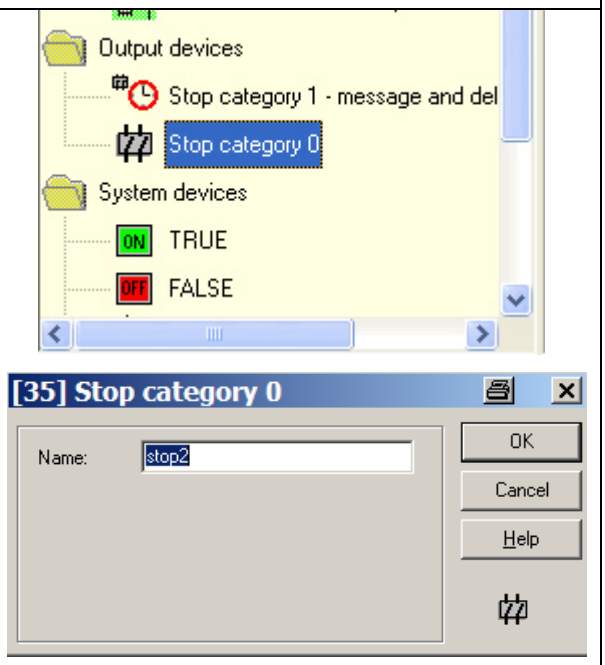
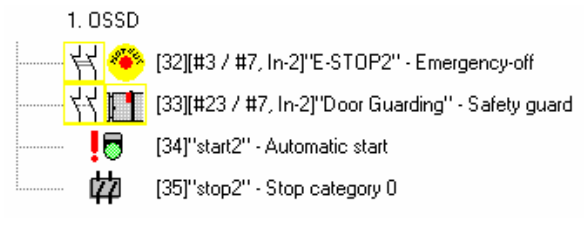
Defining a Stop category	8	Switch-off features are based on stop category 0 (undelayed switch-off). Note: If VSDs have been integrated, a delayed switch-off (stop category 1) can be selected, enabling the controlled ramping down of load disconnection. Confirm with OK.	 
List of all configured functions of the Main safety monitor	9	The complete configuration of the monitor now looks like this. All conditions [32..34] must be true in order for the monitor to be enabled.	
Check of Configuration	10	Click on the check-mark icon to check the configuration. The result will appear in a separate window which will hide itself automatically.	 

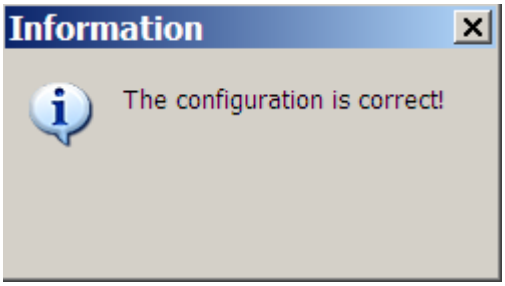
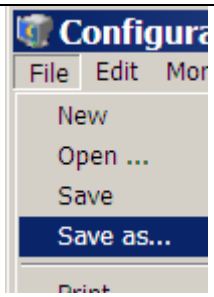
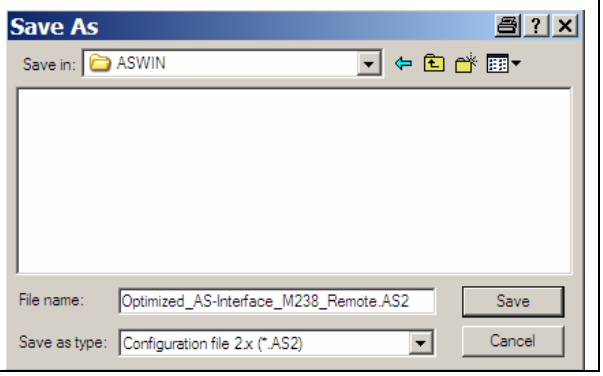
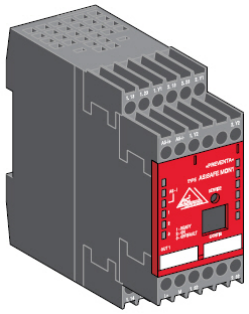
Save configuration of Main Monitor	11	Select **File→ Save as…** to store the Main Safety monitor source: **Optimized_AS-Interface_M238_Main.AS2**	
Starting ASIWIN	1	When the ASIWIN software starts up, the screen opposite will appear, offering you the following options (some of which will be grayed out): • Diagnostics • New configuration • Open configuration • Load configuration from AS-Interface safety monitor If this dialog box does not appear, use **New** or **Open** in the **File menu.**	
Create a New Configuration Setting up the Monitor for the Remote cabinet	2	To create a new configuration, first assign a unique title to your configuration on the **Information about monitor** tab. In this example, an **ASISAFEMON1**, which has just one **OSSD** and basic functionality, is being used.	

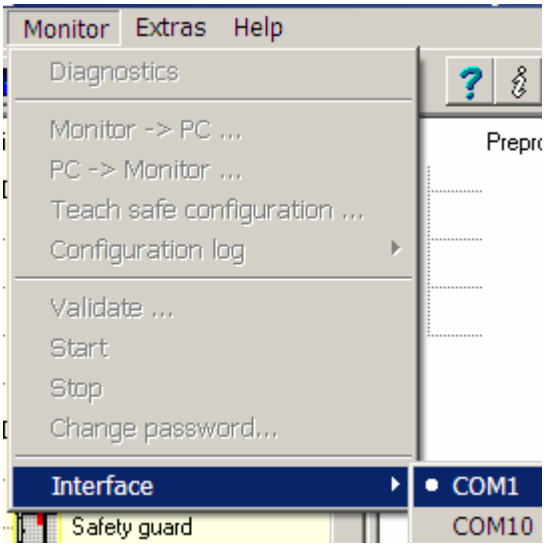
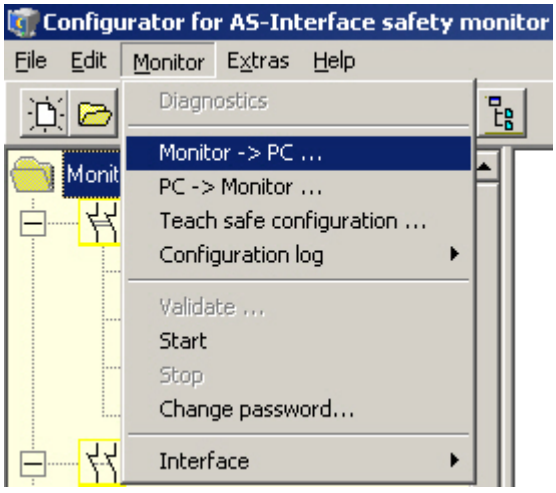
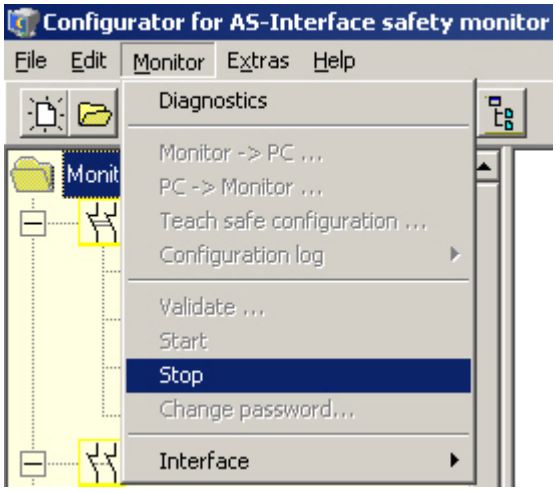
Select the tab Address assignment Fill out the node types safe and standard	3	On the **Information about bus** tab, enter the safe and standard slaves you are using. If you are using standard slaves for acknowledgement and error reset, you will need to enter them here. The slave addresses of Preventa inputs are 2, 3 and 23. The safety monitors for which addresses 1 and 24 have been reserved are not entered here.	
Select the Diagnosis / Service tab Fill out the Monitor base address	4	Enter the address of the safety monitor on the last tab, **Diagnosis / Service.** The address 24 is used for the safety monitor of the Remote cabinet. Confirm with **OK.**	

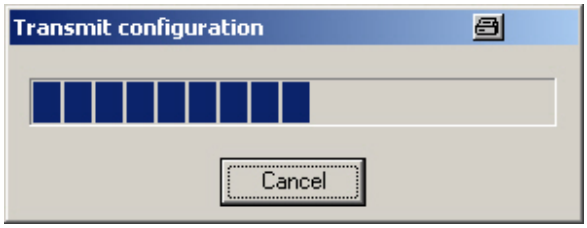
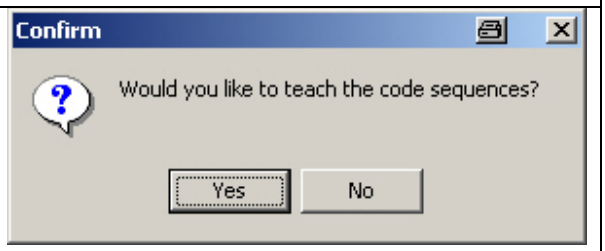
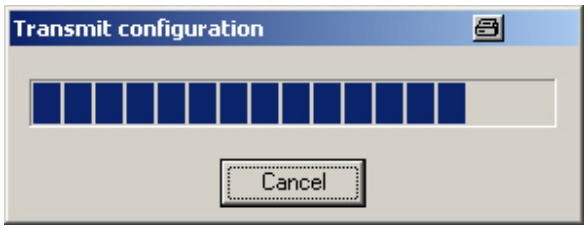
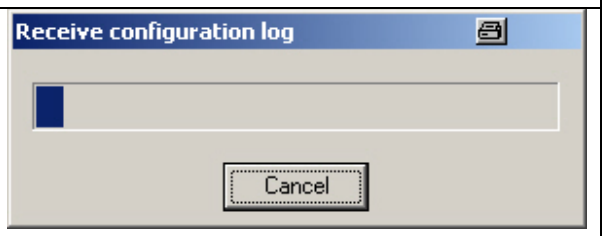
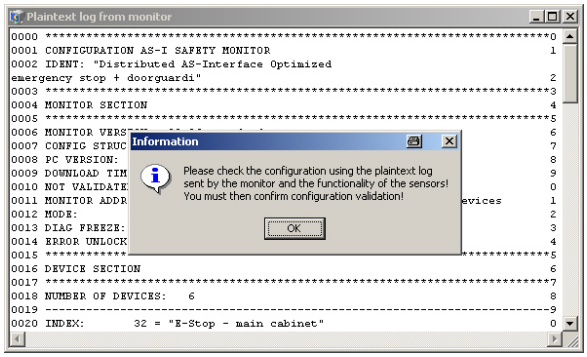
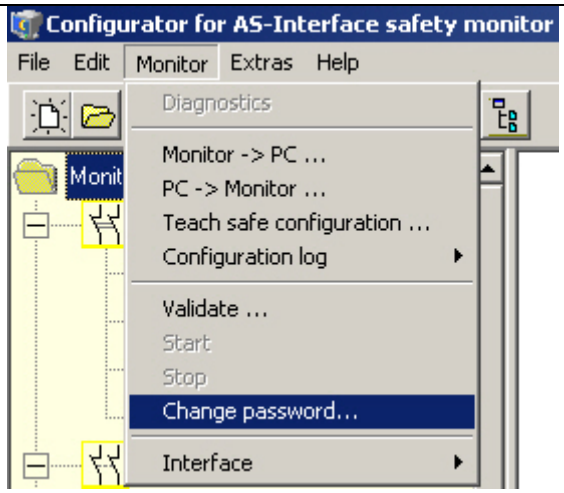
Available Monitor functions	5	The screen on the right will appear for the base controller. A variety of functions can be implemented, these include: • Emergency-off • Safety guard • Module • Feedback loop • Monitored start To use the individual blocks, drag the elements from the yellow area on the left-hand side to the white area on the right-hand side **(1. OSSD).**	
Adding an Emergency-off	6	The following dialog box will appear if you place a forced Emergency-off in the area. In the **Name** field, enter a unique name, which will help you to find the device reliably and without confusion (all device names should be selected on this basis). Next, assign an **Address** to the device; only the addresses you entered as safe slaves when configuring the motor and have not yet used will appear. We use Local acknowledgement. In our example we use the AS-Interface Slave address 7 which depends to one of the XALD Pushbuttons. The Bit address is IN-2. Note: If you activate the Start-up test, when the power supply is restored, the device specified must be forced (triggered) in order to be able to acknowledge the monitor. Confirm with **OK.**	 

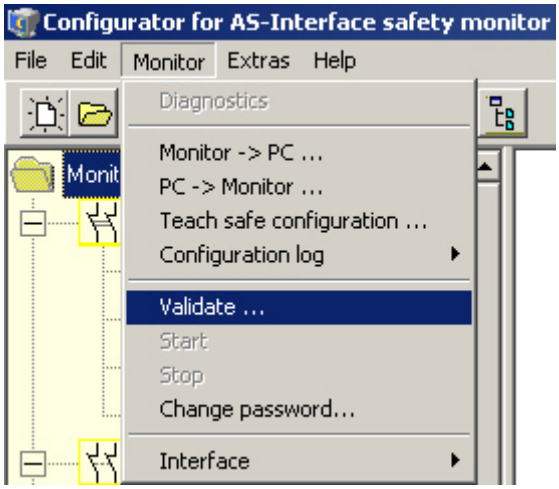
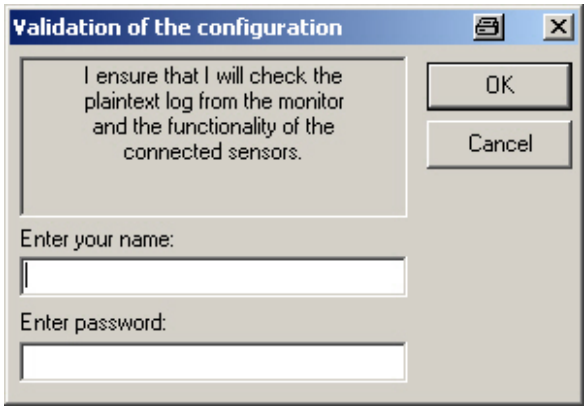
	7	The parameterized Emergency-off function is shown.	
	8	Select **Double channel dependent Emergency-off**	
Adding a Safety Guard	9	The example safety guard is implemented using two roller switches assigned to the same guard. As the switches are positioned to the left and right of the guard, the contacts are not forced directly; rather, there is a dependency between the two switches. On the left of the window, select the **Safety guard** module under Double channel dependent. A period of infinite is entered for the Synchronization time. This time defines the permissible delay of the lagging roller switch. Here the Local acknowledgement is with the Pushbutton on Slave address 7. Confirm with OK.	 

Selecting the Start Condition	10	Once the switches and safety guard monitoring have been added to the configuration, the **Start devices** is added. In the example we use Automatic start. Confirm with OK.	
Defining a Stop category	11	Switch-off features are based on Stop category 0 (undelayed switch-off). Note: If VSDs have been integrated, a delayed switch-off (stop category 1) can be selected, enabling the controlled ramping down of load disconnection. Confirm with OK.	
List of all configured functions of the Main safety monitor	12	The complete configuration of the monitor now looks like this. All conditions [32..35] must be true in order for the monitor to be enabled.	

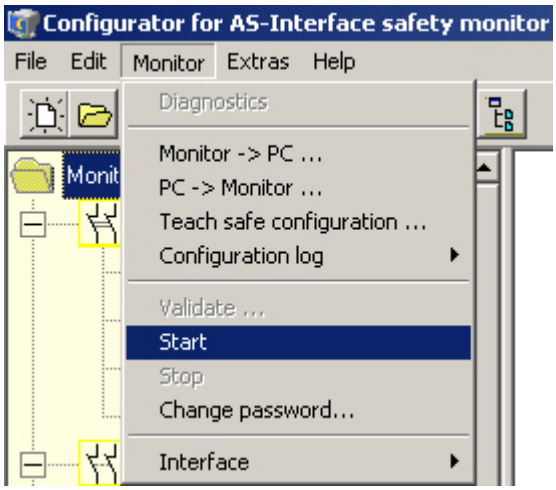
Check of Configuration	13	Click on the check-mark icon to check the configuration. The result will appear in a separate window which will hide itself automatically.	 
Save configuration of Remote Monitor	14	Select **File→ Save as…** to store the Main Safety monitor source: **Optimized_AS-Interface_M238_Remote.AS2**	 
Transferring the configuration to the Monitor	1	The following preconditions now need to be met: • Connect the safety monitor to the power supply. • Connect the configured AS-Interface slaves via the yellow cable. • Wire the monitoring circuit for the load contactors on the monitor. • Connect the PC and monitor via the ASISCPC cable [COM1 (PC) & config port (monitor)].	  ASISCPC ASISAFEMON

2	Before you connect to the monitor, select **Interface** from the Monitor menu to check which communication port has been set. In this case, **COM1** is the only port. There is no need to modify the baud rate or transfer protocol settings.	
3	To transfer a configuration to the monitor, select **PC->Monitor…** from the **Monitor** menu and continue with Step 4. If the four items at the top of the menu list cannot be selected (because they are grayed out), a safety monitor may already have been started (if this is the case, you can stop it by selecting Stop from the Monitor menu). Note: As when starting, the monitor will prompt you to enter a password if you have not connected to the device in the last five minutes or have been offline.	 
4	You will need to enter your password in order for the actions you have requested to be executed. If you are using the safety monitor for the first time, the password will be **SIMON**. Confirm with **OK**.	

	5	The PC will now start to transfer the configuration to the monitor.	
	6	Once the transfer is complete, you will be asked if you would like to teach-in the code sequences. Confirm the prompt with **YES**. The subsequent procedure demonstrates the teaching-in of code tables in order to monitor the state of the safety AS-Interface slaves cyclically.	 
	7	Next, the PC will receive a handshake from the monitor in the form of a log representing the “understood” configuration. This will appear on the screen as a section of plain text and you will be prompted to check the configuration, along with the function of the monitor. Confirm the message with **OK**. Once you have carried out this check, you can click on the close icon (cross) in the top right-hand corner to exit the screen.	 
Change the Password	8	The factory-set default password for new monitors is SIMON. As soon as this password is deactivated on the monitor, you can continue with Step 11. To change the password, select: **Change password…** in the Monitor menu.	

	9	In the top line of the next dialog box that appears, enter the old password SIMON (remember that passwords are case-sensitive). In the middle line, enter your new password and then repeat your entry in the bottom line. Confirm your entries with OK. An error message will appear if the password you have entered is not valid.	
Validate the Monitor	10	To validate the monitor, select: **Validate…** In the Monitor menu. Validation is the last stage before starting the monitor and, functioning virtually as the signature of the responsible programmer, represents the last check prior to first use.	
	11	In the next screen, **Enter your name**, confirm your identity by entering the correct password and confirm your entries with OK.	
	12	Make a note of the information that appears in the next message and keep this in a safe place. Note: Monitors can only be used with valid passwords. However, a generic password can be generated using the field entries.	

Start the Monitor

1	Select **Start** from the Monitor menu to activate the validated safety monitor. It is only when this last setting is made that the signals from the safe slaves are processed and the OSSD can be switched (load connection). Note: Depending on the last direct access via PC to the monitor, you may need to re-enter your password.	

Appendix

Detailed Component List

Main Switch

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
1.0	1	Main switch NSX100F 3pin 36 kA	LV429003	
1.1	1	Contact block TM32D	LV429035	
1.2	1	Terminal cover	LV429515	
1.3	1	Rotary drive with door interface	LV429340	

Emergency Stop

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
2.0	2	Emergency Stop pushbutton	ASISSLB5	
2.1	3	Contactors 9 A, 400 Vac, 24 Vdc	LC1D093BL	
2.2	3	Circuit breaker C60N 1 pole 2 A	23726	
2.3	2	Circuit breaker C60N 2 pole 1 A	23746	
2.4	2	Circuit breaker C60H 2 pole 2 A	25036	
2.5	2	Circuit breaker C60N 2 pole 10 A	23756	
2.6	1	Circuit breaker C60 2 pole 2 A	24443	
2.7	2	Mushroom head, turn to release	ZB4AS844	

Door Guard

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
3.0	1	Safety-Station AS-Interface, 2x M12	ASISSLC2	
3.1	2	Connection cable for Door guard M12, 2 m	XZCP1541L2	
3.2	2	Door guard switch	XCSPL582	
3.3	2	Preventa Safety Monitor	ASISAFEMON1	

Display and indicators

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
4.0	3	Pushbuttons combination red and green	XALS2003H	
4.1	1	Tube with connection	XVBZ02	
4.2	1	Connection element	XVBC21A	
4.3	1	Signal element red	XVBC2B4	
4.4	1	Signal element yellow	XVBC2B5	
4.5	1	Signal element green	XVBC2B3	
4.6	1	lamp white	ZBV-B1	

Automation Components

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
5.0	1	Modicon M238 Logic controller	TM238LDD24DT	
5.1	1	AS-Interface-Master	TM2NOI10M3	
5.2	4	Advantys, AS-Interface for IP 67 I/O	ASI 67FMP43E	
5.3	4	Advantys, AS-Interface for IP 20 I/O	ASI 20MT4I3OSE	
5.4	1	AS-Interface tap-off with stripped ends, AS-Interface & AUX	TCSATV01N2	
5.5	4	AS-Interface tap-off with M12 Connector 90°	TCSATV011F1	
5.6	5	AS-Interface tap-off with M12 Connector	TCSATN011F1	
5.7	2	AS-Interface tap-off with stripped ends	TCSATN01N2	
5.8	2	AS-Interface tap-off connector	TCSATN02V	

Magelis HMI

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
6.0	1	Magelis XBTGT 2220 HMI	XBTGT2220	
6.1	1	Programming cable	XBTZG935	
6.2	1	PLC-HMI Communication cable	XBTZ9008	

Power supply

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
7.0	1	AS-Interface-Power supply	ASIABLM3024	
7.1	1	Disconnect terminal	5711016550	

Drives and Power

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
8.0	6	TeSysU base module reversing 12 A	LU2MB0B	
8.1	6	TeSysU Standard control Unit	LUCA05BL	
8.2	6	TeSysU Coil connection	LU9MRC	
8.3	6	TeSysU AS-Interface-Communication Module	ASILUFC5	
8.4	1	AS-Interface tap-off with stripped ends, AS-Interface & AUX	TCSATV01N2	
8.5	6	AC motor 0.18 kW	MOTOR_380/0, 18kW	
8.6	4	AS-Interface tab	TCSATN01N2	
8.7	1	Contactor	CA4KN22BW3	

Sarel cabinet

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
9.0	1	Cabinet and mounting plate 800 x 600 x 300	83357	
9.1	1	Cabinet and mounting plate 600 x 600 x 300	83330	
9.2	1	Wiring diagram pocket	21322	
9.3	2	Fan with filter; 250 m³; 230 Vac	87901	
9.4	2	Thermostat 1 NC 0-60°C 6 A, 250 Vac	17562	
9.5	2	Cabinet light	21416	

AS-Interface Tools

Hardware Components				
Pos.	Qty.	Description	Part No.	Rev./ Vers.
10.0	1	ASI Addressing Terminal with cable set	ASITERV2SET	**V2**

Software

Software Components				
Pos.	Qty.	Description	Part Number	Rev./ Vers.
11.0	1	SoMachine (Includes Vijeo Designer)	MSDCHNSFUV20	V2.0
11.1	1	ASI PC-Software Safety MonitorV2	ASISWIN2	V2.03

Component Protection Classes

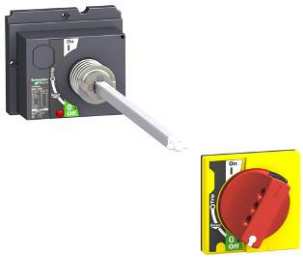
Recommended installation locations/ Protection class

Components	In the Field / on Site			Cabinet		
				Front		inside
	IP54	IP65	IP67	IP55	IP65	IP 20
Emergency Stop installation box				X		X
TeSys contactors						X
Circuit breakers						X
Phaseo power supply unit						X
TeSysU motor starters						X
Modicon M238 Logic controller including expansion modules						X
Magelis display terminal					X	X
Harmony pushbuttons in housing		X				
AS-Interface – I/O modules ASI20MT4I3OSE						X
AS-Interface – I/O modules ASI67FMP43E			X			
AS-Interface safety monitor						X
AS-Interface Emergency Stop slave ASISSLB4						X
Mushroom attachment/switch for ASISSLB4					IP66	
AS-Interface Emergency-off slave ASISSC1/C2			X			
Preventa safety limit switch XCSM-PL			X			

Component Features

Components

Compact NSX main switch



Compact NSX rotary switch disconnectors from 12 to 175 A are suitable for on-load making and breaking of resistive or mixed resistive and inductive circuits where frequent operation is required. They can also be used for direct switching of motors in utilization categories AC-3 and DC-3 specific to motors.

- 3-pole rotary switch disconnectors, 12 to 175 A
- Padlockable operating handle (padlocks not supplied)

Degree of protection IP 65

Preventa Safety Module: ASISAFEMON1

Main technical characteristics:

For monitoring	Emergency stop, Safety switches, Safety light curtains
Max. category accord. EN954-1	4
AS-Interface profile	7.F
Power supply AC/DC	24V ± 15%
Consumption on AS-Interface Line	44 mA
Operating temperature	- 20…+60 °C
Indicators	5 LED's
Number of Safety circuits	2 N/O
Number of additional circuits	1 solid-state output for signaling to controller
Response time on input opening	< 40 ms
Fuse protection	External, with max. of 4 A MT
Dimensions (mm)	45 x 104 x 120
Degree of protection	IP20

Phaseo Power Supply Unit: ASIABLM3024

- 100...120 Vac and 200...500 Vac input
- Two separate independent Outputs: 30 Vdc (AS-Interface line supply) and 24 Vdc Output
- 2.4 A (30 Vdc) and 3 A (24 Vdc) output
- Diagnostic relay

Magelis Display Terminal: XBTGT2220

- Sensor screen (STN-Technology) with 24 Vdc power supply
- Brightness and Contrast adjustment
- Communication via Uni-Telway and Modbus.
  Communication via Ethernet TCP/IP is also available in specific models
- Flat Profile
- Memory expansion for application program
- Temperature range: 0..+ 50 °C
- Certificates: UL, CSA

Modicon M238 Logic controller

The M238 is powered with 24 Vdc, offer:

- 14 x 24 Vdc inputs including 8 fast inputs, dedicated to special functions such as HSC high-speed counting
- 10 x 24 Vdc solid state outputs including 4 fast outputs, dedicated to special functions such as counting, PWM and PTO
- An RS 232/RS 485 serial link (ASCII or Modbus protocol).
- A Modbus RS 485 serial link mainly dedicated to connection of a Human/Machine interface terminal (link providing a 5 V power supply for a Magelis Small Panel XBT NP00/R400/RT500)
- Expand the I/O count by adding up to 7 expansion modules. The following modules are available:
  - Discrete TM2 DDI/DDO/DMM/DRA
  - Analog TM2 AMI/ALM/ARI/AMO/AVO/AMM
  - High-speed counter TM200 HSC210DT/DF
  - AS-Interface Master TWDNOI10M3 (max. 2)

SoMachine OEM Machine Programming Software: MSDCHNSFUV20

SoMachine is the OEM solution software for developing, configuring and commissioning the entire machine in a single software environment, including logic, motion control, HMI and related network automation functions.

SoMachine allows you to program and commission all the elements in Schneider Electric's Flexible and Scalable Control platform, the comprehensive solution-oriented offer for OEMs, which helps you achieve the most optimized control solution for each machine's requirements.

Flexible and Scalable Control platforms include:

Controllers:

HMI controllers:
- Magelis XBTGC HMI controller
- Magelis XBTGT HMI controller
- Magelis XBTGK HMI controller

Logic controllers:
- Modicon M238 Logic controller
- Modicon M258 Logic controller

Motion controller
- Modicon LMC058 Motion controller

Drive controller:
- Altivar ATV-IMC Drive controller

HMI:

HMI Magelis graphic panels:
- XBTGT
- XBTGK

SoMachine is a professional, efficient, and open software solution integrating Vijeo-Designer.
It integrates also the configuring and commissioning tool for motion control devices.

It features all IEC 61131-3 languages, integrated field bus configurators, expert diagnostics and debugging, as well as outstanding capabilities for maintenance and visualization.

SoMachine provides you:
- One software package
- One project file
- One cable connection
- One download operation

ASISWIN2 - AS-Interface safety monitor configuration software

- Multilingual EN / FR / DE / ES / IT / PT
- For use with ASISAFEMON1/2, ASISAFEMON1B/2B
- Media CD-ROM PC
- Environment Microsoft Windows

Functions for ASISAFEMON1	**"Basic"**
Number of functional components at logic level	32
OR gates (inputs)	2
AND gates (inputs)	no
Safe time function, switch-on and switch-off delay	no
Function "button"	no
Safety guard/module with debouncing	no
Safety guard with lock	no
Deactivation of functional components	yes
Error release	yes
Diagnosis stop	yes
Support of A/B technology for non-safe slaves	yes
New functional components (flip-flop, pulse if pos. edge, etc.)	no
Dummy device (NOP)	no

Contact

Publisher	**Process & Machine Business** **OEM Application & Customer Satisfaction** Schneider Electric Automation GmbH Steinheimer Strasse 117 D - 63500 Seligenstadt Germany
Homepage	http://www.schneider-electric.com/sites/corporate/en/home.page
As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.	